Documentation 8.2

ZABBIX

05.11.2026

Contents

Podręcznik użytkownika **5**

Witamy w dokumentacji Zabbix **5**

Rozpocznij pracę z Zabbix . . . 5
Zabbix Cloud . . . 5
Aplikacja mobilna . . . 5
Centrum dewelopera . . . 5
Społeczność i inne zasoby . . . 5
Instalacja i pierwsze kroki . . . 6
Instalacja interfejsu webowego . . . 6
Przewodniki szybkiego startu . . . 11
Instalacja: dodatkowe przewodniki . . . 85
Aktualizacja . . . 165
Znane problemy . . . 182
Wymagania . . . 190
Co nowego i uwagi dotyczące aktualizacji: 8.0 . . . 200
Uwagi dotyczące aktualizacji . . . 200
Zmiany niekompatybilne wstecz . . . 201
Inne . . . 201
Co nowego i uwagi dotyczące aktualizacji: 8.0.x . . . 201
Urządzenie Zabbix . . . 201
Procesy . . . 205
Serwer . . . 207
Proxy . . . 234
Agent . . . 280
Agent 2 . . . 310
Java gateway . . . 373
Nadawca . . . 377
Pobierz . . . 379
JS . . . 380
Usługa sieciowa . . . 381
Adapter mostu . . . 384
Serwer MCP . . . 388
Zmienne środowiskowe . . . 395
Konfiguracja . . . 397
Hosty i grupy hostów . . . 408
Pozycje . . . 426
Wykrywanie problemów za pomocą wyzwalaczy . . . 652
Wizualizacja . . . 704
Szablony i grupy szablonów . . . 729
Szablony dostępne od razu po instalacji . . . 729
Powiadomienia po wystąpieniu zdarzeń . . . 743
Makra . . . 795
Użytkownicy i grupy użytkowników . . . 810
Zaplanowane raporty . . . 820
Eksport danych . . . 824
Eksport/import konfiguracji . . . 833
Wyrażenia regularne . . . 900
Zdarzenia i rozwiązanie . . . 906
Potwierdzanie i tłumienie problemów . . . 908
Ręczne zamykanie problemów . . . 911

Okresy konserwacji . . . 913
Tagowanie . . . 919
Complex event processing . . . 921
Korelacja zdarzeń . . . 925
Odkrywanie . . . 932
Odkrywanie sieci . . . 932
Automatyczna rejestracja aktywnego agenta . . . 941
Odkrywanie niskiego poziomu . . . 945
Monitorowanie maszyn wirtualnych . . . 968
Klucze pozycji monitorowania VMware . . . 970
Klucze wykrywania maszyn wirtualnych . . . 988
Przykłady JSON dla pozycji VMware . . . 992
Przykład konfiguracji monitorowania VMware . . . 997
Monitorowanie sieci web . . . 1002
Pozycje monitorowania sieci web . . . 1011
Scenariusz z życia wzięty . . . 1012
Monitorowanie usług . . . 1019
Drzewo usług . . . 1020
SLA . . . 1024
Przykład konfiguracji . . . 1025
Ulepszenie monitorowania usług . . . 1030
Szyfrowanie i bezpieczeństwo . . . 1030
Szyfrowanie przy użyciu certyfikatów . . . 1039
Szyfrowanie przy użyciu kluczy współdzielonych z wyprzedzeniem . . . 1046
Bezpieczne połączenie z bazą danych . . . 1049
Bezpieczne połączenie między frontendem a serwerem . . . 1061
Kontrola dostępu . . . 1065
Kryptografia . . . 1069
Zabezpieczanie serwera WWW . . . 1070
Przechowywanie sekretów . . . 1073
Zaufanie do certyfikatu TLS w Google Chrome . . . 1082
Najlepsze praktyki dotyczące bezpieczeństwa . . . 1083
Interfejs webowy . . . 1084
Menu . . . 1084
Sekcje frontend . . . 1093
Ustawienia użytkownika . . . 1364
Globalne wyszukiwanie . . . 1369
Tryb konserwacji frontend . . . 1371
Parametry strony . . . 1372
Definicje . . . 1375
Dodatkowe języki frontend . . . 1375
Tworzenie własnego motywu . . . 1376
Tryb debugowania . . . 1377
Pliki cookie używane przez Zabbix . . . 1378
Strefy czasowe . . . 1379
Resetowanie hasła . . . 1380
Okres czasu i selektory hostów . . . 1381
Parametry konfiguracji: frontend Zabbix . . . 1382
API . . . 1389
Odniesienie do metody . . . 1394
Dodatek 1. Komentarz referencyjny . . . 2277
Załącznik 2. Zmiany od 7.4 do 8.0 . . . 2283
Załącznik 3. Zmiany w wersji 8.0 . . . 2283
Rozszerzenia . . . 2283
Moduły ładowalne . . . 2285
Moduły frontend . . . 2294
Aneksy . . . 2295
Instalacja i konfiguracja . . . 2295
Pozycje . . . 2342
Obsługiwane funkcje . . . 2383
Makra . . . 2383
Zgodność wersji . . . 2419
Składnia okresu czasu . . . 2421

Symbole jednostek . . . 2421
Wykonywanie poleceń . . . 2423
Przykłady ucieczki znaków . . . 2424
Czym jest Zabbix . . . 2425

Zabbix Cloud **2430**

Zabbix Cloud **2430**
Konfiguracja i zarządzanie Zabbix Cloud . . . 2430
Dowiedz się więcej o Zabbix Cloud . . . 2430
Wdróż Zabbix w chmurze . . . 2430
Konfiguracja węzła . . . 2439
Przegląd . . . 2440
Karta Przegląd . . . 2441
Karta filtrów dostępu . . . 2444
Karta Encryption . . . 2445
Karta Kopie zapasowe . . . 2445
Karta historii . . . 2446
Karta Maintenance . . . 2446
Karta „Upgrade” . . . 2447
Dodawanie użytkowników . . . 2449
Wprowadzenie . . . 2449
Dodawanie użytkowników do organizacji . . . 2449
Dodawanie użytkowników do węzłów . . . 2450
Kluczowe różnice w Zabbix Cloud . . . 2451
Wprowadzenie . . . 2451
Dziennik audytu . . . 2453
Konfigurowanie proxy Zabbix Cloud . . . 2455

Aplikacja mobilna **2463**

Aplikacja mobilna **2463**
Skonfiguruj Zabbix dla aplikacji . . . 2463
Dowiedz się więcej o aplikacji . . . 2463
Konfigurowanie Zabbix na potrzeby aplikacji . . . 2464
Przegląd aplikacji . . . 2465
Bezpieczny dostęp mobilny . . . 2473

Centrum dewelopera **2474**

Centrum deweloperskie **2474**
Twórz na Zabbix . . . 2474
Tworzenie wtyczki . . . 2474
Interfejsy wtyczek . . . 2480
Biblioteka Python dla Zabbix . . . 2482
Instalacja . . . 2482
Przewodnik szybkiego startu . . . 2482
Korzystanie z API Zabbix . . . 2486
Zbieranie danych z agenta Zabbix . . . 2489
Wysyłanie danych do serwera lub proxy Zabbix . . . 2490
Rejestrowanie debugowania . . . 2494
Moduły . . . 2495
Struktura plików modułu . . . 2495
Widżety . . . 2503
Poradniki . . . 2511
Przykłady . . . 2537
Zmiany w tworzeniu rozszerzeń . . . 2537

Strony podręcznika Zabbix **2538**
zabbix_agent2 . . . 2538
zabbix_agentd . . . 2539
zabbix_get . . . 2541
zabbix_js . . . 2543
zabbix_proxy . . . 2544
zabbix_sender . . . 2546

zabbix_server 2551
zabbix_web_service 2553

Informacja o prawach autorskich **2554**

Podręcznik użytkownika

Witamy w dokumentacji Zabbix

Twoje podstawowe źródło informacji do pracy z monitoringiem Zabbix — od podstawowych wdrożeń po zaawansowane konfiguracje. Ten podręcznik obejmuje wszystko, co jest potrzebne do instalacji, konfiguracji i obsługi Zabbix.

Co nowego w Zabbix 8.0

Rozpocznij pracę z Zabbix

Instalacja Instrukcje krok po kroku dotyczące instalacji Zabbix na wybranej platformie, obejmujące różne systemy operacyjne i konfiguracje baz danych.

Wymagania Lista obsługiwanych platform i wymagań wstępnych dotyczących oprogramowania, która pomoże Ci przygotować środowisko do pomyślnego wdrożenia Zabbix.

Przewodniki szybkiego startu Zwięzłe, nastawione na wykonanie zadania instrukcje krok po kroku, które przeprowadzą Cię przez podstawy — od pierwszych kroków konfiguracji po otrzymanie pierwszego alertu o problemie.

Zabbix Cloud

Rozpocznij korzystanie z Zabbix Cloud Instrukcje krok po kroku, które pomogą szybko uruchomić instancję Zabbix Cloud, w tym przeprowadzić wstępną konfigurację i podłączyć monitorowane urządzenia.

Zabbix Cloud a on-premises Porównaj różnice w zarządzaniu, skalowalności i utrzymaniu między tymi dwoma wdrożeniami i dowiedz się, które z nich może najpłynniej wspierać Twój przepływ pracy.

Poznaj Zabbix Cloud Zapoznaj się z przeglądem Zabbix Cloud, w tym z jego kluczowymi funkcjami, korzyściami oraz tym, co obejmuje subskrypcja. Odpowiedzi na często zadawane pytania dotyczące konfiguracji i użytkowania można znaleźć w FAQ.

Aplikacja mobilna

Pobierz aplikację Zainstaluj aplikację mobilną Zabbix (wkrótce dostępną) na swoim urządzeniu i rozpocznij monitorowanie swojego środowiska Zabbix w dowolnym miejscu.

Skonfiguruj Zabbix dla aplikacji Zainstaluj adapter Zabbix bridge, aby umożliwić urządzeniom mobilnym łączenie się z Zabbix. Następnie dodaj swoje urządzenie i włącz powiadomienia push.

Przegląd aplikacji Poznaj podstawy korzystania z aplikacji mobilnej Zabbix.

Centrum dewelopera

Moduły frontend Przewodniki i materiały referencyjne dotyczące tworzenia niestandardowych modułów, które rozszerzają lub modyfikują frontend Zabbix, aby dostosować go do konkretnych przypadków użycia.

Widgety Omówienie struktury i logiki widgetów wraz z instrukcjami tworzenia niestandardowych elementów pulpitu dostosowanych do Twoich potrzeb.

Wtyczki Przegląd sposobu tworzenia i zarządzania wtyczkami Zabbix w celu rozszerzenia funkcjonalności lub integracji z systemami zewnętrznymi.

Społeczność i inne zasoby

Fora Zabbix Miejsce do wymiany pomysłów, znajdowania rozwiązań i dzielenia się doświadczeniami na wszystkich poziomach zaawansowania w Zabbix.

Blog Zabbix Aktualności, samouczki i studia przypadków przygotowane przez zespół Zabbix oraz współtwórców społeczności.

Filmy instruktażowe Biblioteka filmów zawierająca demonstracje, dyskusje i wskazówki, które pomogą Ci w pełni wykorzystać możliwości Zabbix.

Instalacja i pierwsze kroki

Następująca sekwencja przedstawia proces konfiguracji Zabbix:

1. Sprawdź, czy wymagane oprogramowanie jest zainstalowane i uruchomione.
2. Zainstaluj Zabbix.
3. Zainstaluj interfejs webowy Zabbix.
4. Rozpocznij monitorowanie.

Szczegóły dotyczące każdego kroku znajdują się poniżej.

Wymagane oprogramowanie Zabbix dobrze współpracuje z następującym oprogramowaniem (często nazywanym stosem LAMP):

- **System operacyjny Linux:** Ubuntu, Debian, CentOS lub inne
- **Serwer WWW:** Apache lub NGINX
- **Serwer bazy danych:** MySQL/MariaDB lub PostgreSQL
- **PHP**

W większości nowoczesnych systemów Linux PHP może być już zainstalowane. W takim przypadku może być konieczne skonfigurowanie tylko serwera bazy danych i serwera WWW.

Więcej szczegółowych informacji, takich jak obsługiwane wersje oprogramowania, można znaleźć w sekcji Wymagane oprogramowanie.

Instalacja Zabbix **Najłatwiejszym** sposobem instalacji Zabbix jest użycie **pakietów dystrybucyjnych** (alternatywnie można skorzystać z kontenerów). Pakiety instalują wymagane zależności oprogramowania.

Dokładne instrukcje instalacji dostosowane do systemu operacyjnego są dostępne na **stronie internetowej Zabbix**.

Wybierz swoją platformę i postępuj zgodnie z podanymi instrukcjami. Na początek zaleca się zainstalowanie serwera Zabbix, agenta Zabbix 2 oraz frontendu (interfejsu webowego). Później można dodać proxy Zabbix i kolejne agenty.

Częścią procesu instalacji jest utworzenie bazy danych (do przechowywania danych konfiguracyjnych i historycznych) oraz zaimportowanie do niej danych początkowych.

Instalacja interfejsu webowego Zabbix Wykonaj kroki **kreatora instalacji** interfejsu webowego.

Zostanie wyświetlona prośba o podanie hasła do bazy danych (utworzonego w poprzednim kroku), aby frontend mógł komunikować się z bazą danych.

Rozpoczęcie monitorowania Zaloguj się do interfejsu webowego Zabbix.

Skorzystaj z **przewodników szybkiego startu**, aby rozpocząć monitorowanie popularnych zasobów (Linux, Apache, MySQL lub innych). Przewodniki szybkiego startu wyjaśniają, z którymi szablonami należy rozpocząć monitorowanie popularnych zasobów.

Możesz rozpocząć od przewodnika szybkiego startu dotyczącego monitorowania serwera Linux.

Instalacja interfejsu webowego

Ta sekcja zawiera instrukcje dotyczące instalacji frontend Zabbix. Wykonaj te kroki po zainstalowaniu backendu Zabbix.

Frontend Zabbix jest napisany w języku PHP i wymaga obsługiwanego serwera webowego, takiego jak Apache lub NGINX, z odpowiednio skonfigurowanym PHP. Informacje o wymaganym oprogramowaniu i bibliotekach znajdziesz w sekcji wymagań frontend.

Aby używać w frontend Zabbix języka innego niż angielski, jego locale musi być zainstalowane na serwerze webowym wraz z rozszerzeniem PHP gettext. Pamiętaj, że zmiana języka może zmienić format godziny i daty.

Aby skonfigurować SSL dla frontend Zabbix, zobacz Konfigurowanie SSL dla frontend Zabbix.

Kreator instalacji

Aby zainstalować frontend, otwórz adres URL frontend Zabbix w przeglądarce:

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

Ekran powitalny

Kreator instalacji frontend otwiera się na ekranie powitalnym.

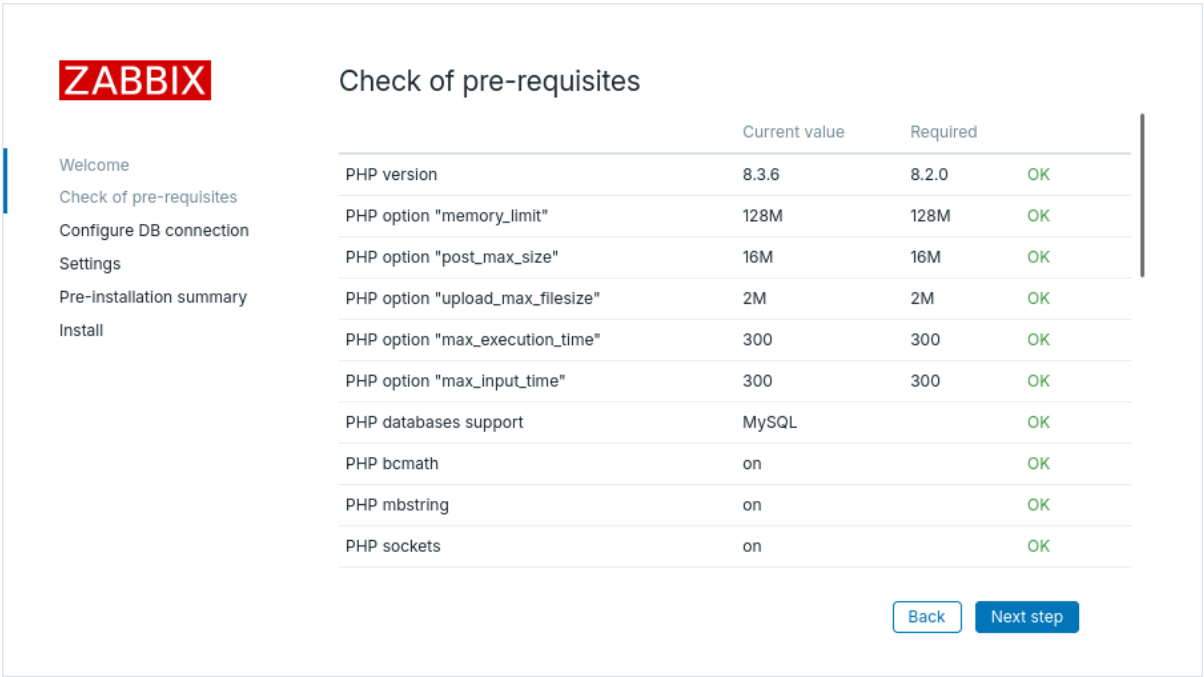


Domyślnym językiem całego systemu jest angielski (en_US), z amerykańskim formatem godziny i daty. Aby zmienić domyślny język całego systemu, wybierz żądany język z menu rozwijanego *Default language*.

Kliknij przycisk *Next step*, aby kontynuować.

Sprawdzanie wymagań wstępnych

Upewnij się, że wszystkie obowiązkowe wymagania wstępne frontend Zabbix są spełnione.



Listę wymaganych i opcjonalnych rozszerzeń PHP, wymaganą wersję PHP oraz obsługiwane bazy danych znajdziesz w sekcji Wymagania. Pamiętaj, że niespełnione opcjonalne wymagania wstępne są wyświetlane ze stanem *Ostrzeżenie*, a proces konfiguracji może być kontynuowany nawet wtedy, gdy nie są spełnione.

Wymagane wartości dyrektyw konfiguracji PHP w pliku `php.ini` używanym przez serwer WWW:

```
memory_limit = 128M ; Wartość minimalna
post_max_size = 16M ; Wartość minimalna
upload_max_filesize = 2M ; Wartość minimalna
```

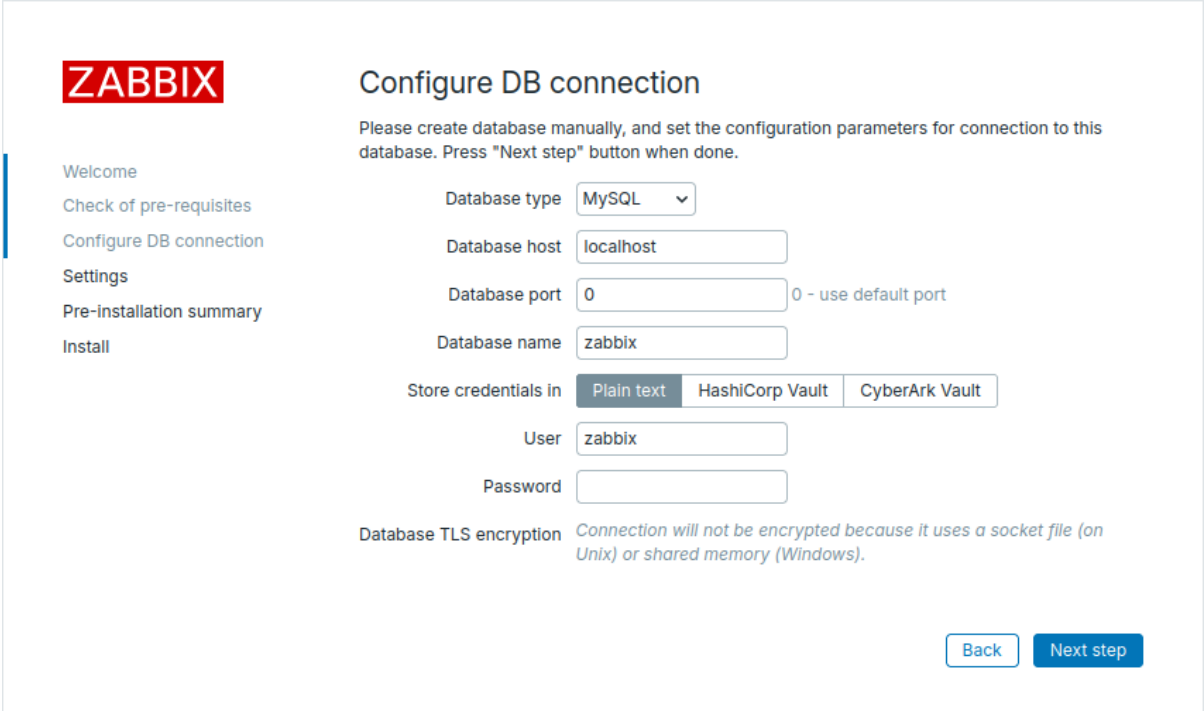
```
max_execution_time = 300 ; Wartość minimalna; dozwolone są również wartości 0 i -1
max_input_time = 300 ; Wartość minimalna; dozwolone są również wartości 0 i -1
session.auto_start = 0 ; Musi być wyłączone
arg_separator.output = "&" ; Dozwolone jest również "&amp;"
```

Attention:
Jeśli użytkownik lub grupa serwera WWW musi zostać zmieniona, upewnij się, że nowy użytkownik lub grupa ma wymagane uprawnienia do katalogu sesji PHP; w przeciwnym razie konfiguracja Zabbix może nie być możliwa do kontynuowania.

Jeśli wszystkie obowiązkowe wymagania wstępne są spełnione, przejdź do następnego kroku.

Konfiguracja połączenia z bazą danych

Wprowadź dane potrzebne do połączenia z bazą danych.



Jeśli pole *Host bazy danych* pozostanie puste, zostanie użyte domyślne gniazdo domeny Unix.

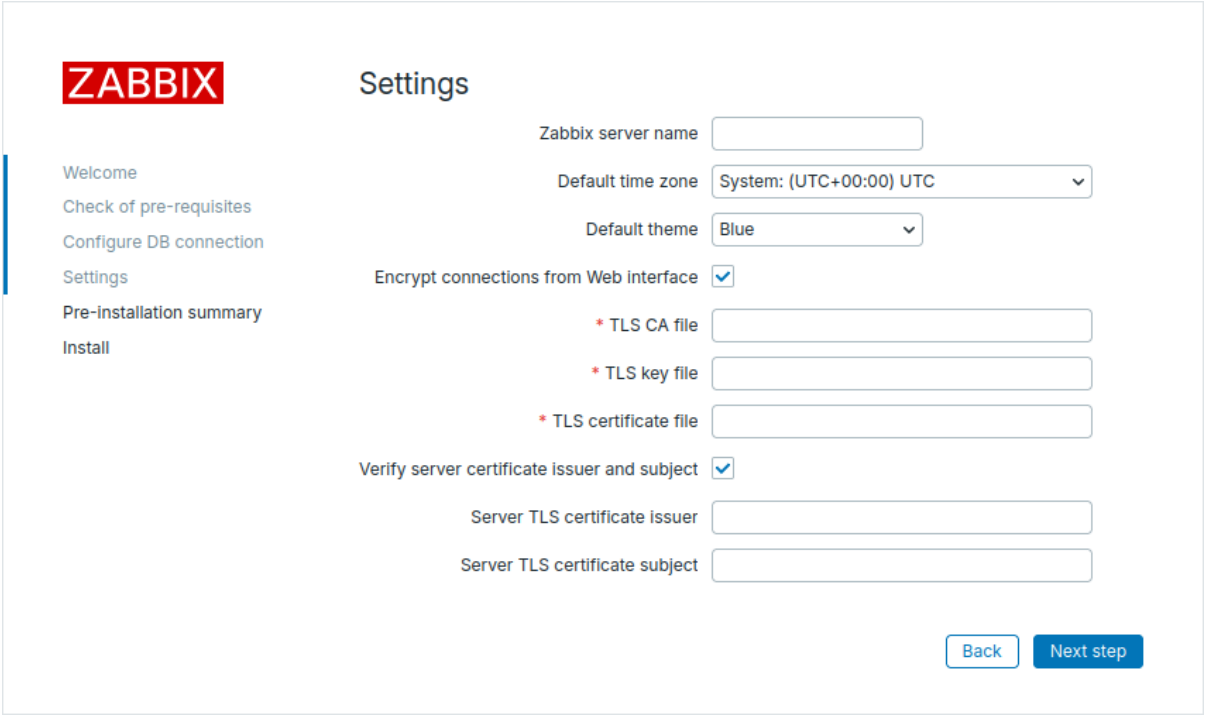
W przypadku MySQL wpisanie `localhost` w polu *Host bazy danych* również powoduje połączenie za pośrednictwem domyślnego gniazda domeny Unix. Kreator nie udostępnia opcji konfiguracji gniazda bazy danych. Aby użyć niestandardowego gniazda, skonfiguruj je w ustawieniach serwera Zabbix (na przykład za pomocą parametru DBSocket= w pliku konfiguracyjnym serwera Zabbix (`zabbix_server.conf`)). Dzięki temu frontend pozostaje spójny z ustawieniami połączenia serwera z bazą danych.

W przypadku PostgreSQL pole *Host bazy danych* może zawierać wiele wpisów `host:port` oddzielonych przecinkami. Ciąg jest dzielony na części, a każdy port jest sprawdzany w taki sam sposób jak pole *Port bazy danych*. Po pomyślnej walidacji ciąg jest analizowany na hosty i porty w celu utworzenia parametrów połączenia. Jeśli dla hosta na rozdzielonej przecinkami liście nie zdefiniowano portu, jako port domyślny używana jest wartość pola *Port bazy danych*. Jeśli określono ścieżkę gniazda (na przykład `/var/run/pgbouncer`), używane jest to gniazdo domeny Unix.

Aby skonfigurować bezpieczne połączenie z bazą danych, zaznacz pole wyboru *Szyfrowanie TLS bazy danych* i przejdź do sekcji Frontend do bazy danych, aby skonfigurować pola. Pole wyboru *Szyfrowanie TLS bazy danych* jest wyłączone w następujących przypadkach: - W przypadku MySQL wartość *Host bazy danych* to *localhost*. - W przypadku PostgreSQL wartość pola *Host bazy danych* zaczyna się od ukośnika lub pole jest puste.

Aby skonfigurować Zabbix do pobierania sekretów z HashiCorp Vault, wybierz *HashiCorp Vault* w polu *Przechowuj dane uwierzytelniające w* i przejdź do sekcji Konfiguracja HashiCorp, aby wypełnić rozwinięte pola. Aby skonfigurować Zabbix do pobierania sekretów z CyberArk Vault, wybierz *CyberArk Vault* w polu *Przechowuj dane uwierzytelniające w* i przejdź do sekcji Konfiguracja CyberArk, aby wypełnić rozwinięte pola.

Ustawienia



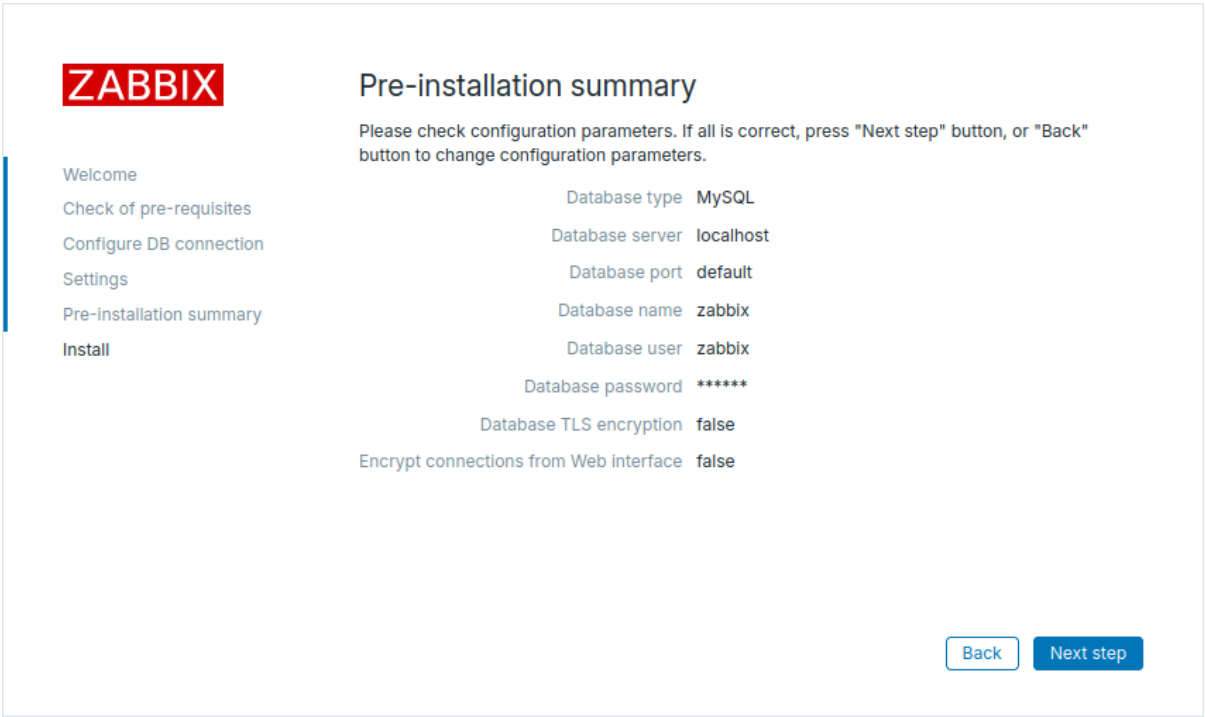
Wprowadzenie nazwy serwera Zabbix jest opcjonalne. Jeśli jednak zostanie ona podana, będzie wyświetlana na pasku menu i w tytułach stron.

Ustaw domyślną strefę czasową i motyw dla frontendu.

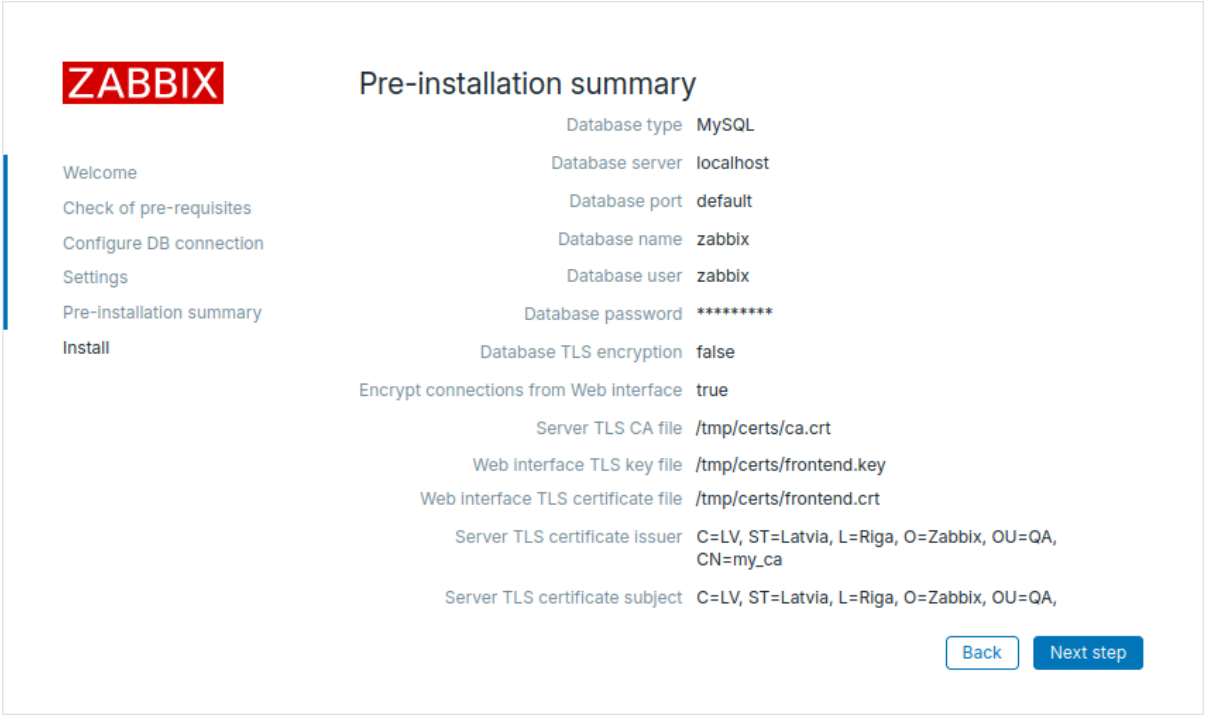
Aby skonfigurować połączenia TLS między frontendem Zabbix a serwerem Zabbix, zaznacz pole wyboru *Szyfruj połączenia z interfejsu internetowego* i przejdź do konfigurowania połączenia TLS, aby wypełnić rozwinięte pola.

Podsumowanie przed instalacją

Przejrzyj podsumowanie ustawień.



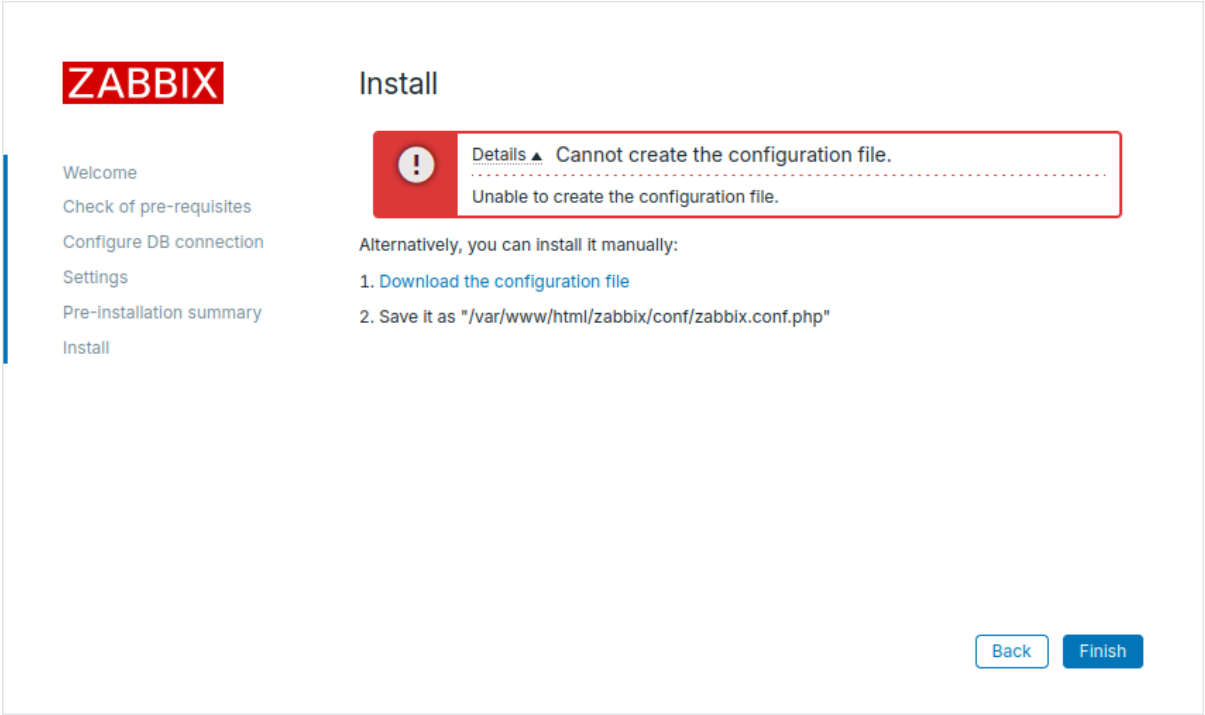
Jeśli skonfigurowano TLS, kreator wyświetli szczegółowe informacje.



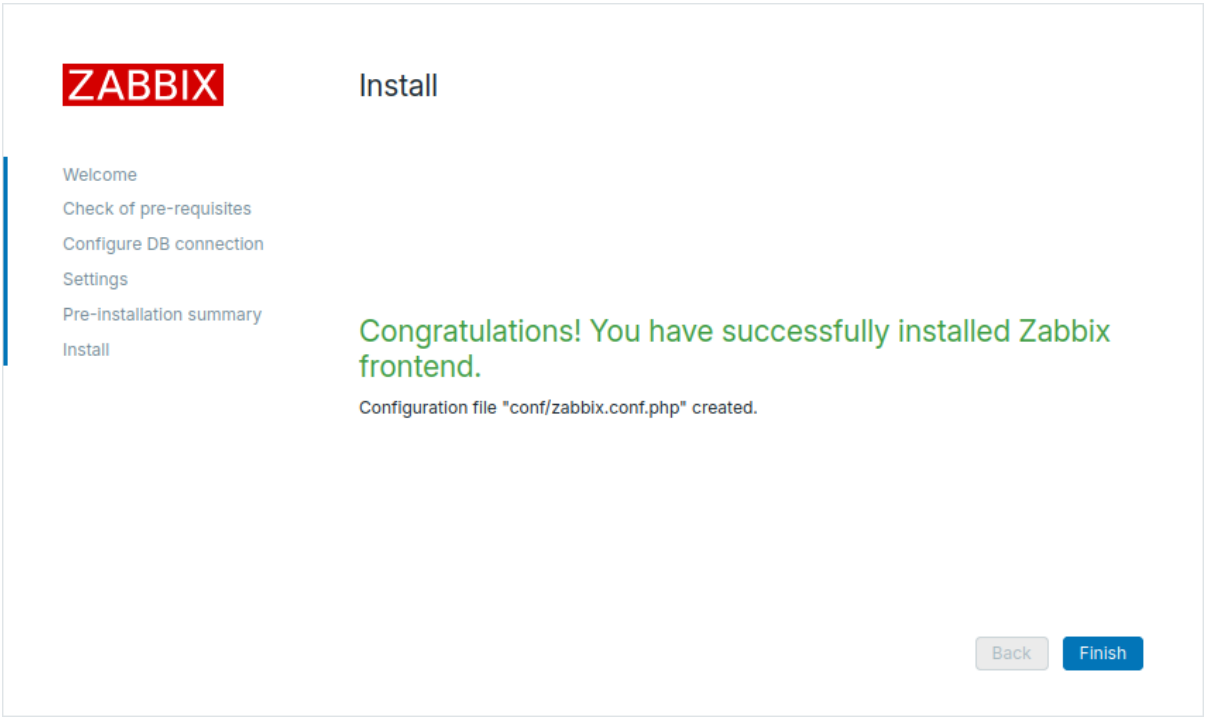
Instalacja

Jeśli użytkownik serwera WWW ma uprawnienia do zapisu w katalogu `conf/`, plik konfiguracyjny zostanie zapisany automatycznie i można od razu przejść do następnego kroku.

Jeśli użytkownik serwera WWW nie ma uprawnień do zapisu w katalogu `conf/`, należy pobrać plik konfiguracyjny i umieścić go w katalogu `conf/` w katalogu dokumentów HTML serwera WWW, do którego skopiowano pliki PHP Zabbix.

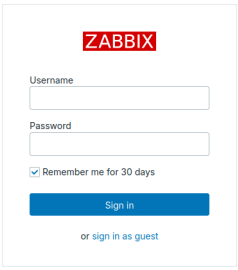


Zakończ instalację.



Logowanie

Zaloguj się przy użyciu domyślnych danych uwierzytelniających: - Nazwa użytkownika: *Admin* - Hasło: *zabbix*



Plik konfiguracyjny frontendu

Po instalacji można ponownie skonfigurować frontend Zabbix, aktualizując parametry w pliku konfiguracyjnym frontendu `zabbix.conf.php`.

Domyślna ścieżka do pliku konfiguracyjnego zależy od systemu operacyjnego i metody instalacji Zabbix:

- W przypadku instalacji z pakietów: `/usr/share/zabbix/ui/conf/zabbix.conf.php`
- W przypadku instalacji ze źródeł: `/var/www/html/zabbix/conf/zabbix.conf.php`
- W przypadku appliance: `/etc/zabbix/web/zabbix.conf.php`

Szczegółowe informacje dotyczące konfigurowania frontendu Zabbix zawiera sekcja opcje pliku konfiguracyjnego.

Przewodniki szybkiego startu

Po zainstalowaniu Zabbixa i jego interfejsu webowego możesz rozpocząć monitorowanie.

Skorzystaj z tych krótkich przewodników, aby rozpocząć monitorowanie popularnych zasobów:

- Monitorowanie serwera Linux
- Monitorowanie serwera webowego Apache
- Monitorowanie serwera MySQL

Przewodniki szybkiego startu koncentrują się na używaniu oficjalnych szablonów Zabbix do monitorowania. Szablon to wstępnie skonfigurowany zestaw elementów monitorowania. Szablon umożliwia skalowanie monitorowania — możesz przetestować szablon na jednym monitorowanym obiekcie (urządzeniu, aplikacji itp.), dostosować go, a następnie zastosować do wielu podobnych monitorowanych obiektów.

Szablony dobrze sprawdzają się w monitorowaniu od razu po instalacji, ale możesz również chcieć dostosować ich elementy.

Aby lepiej zrozumieć podstawowe elementy składające się na szablon, zapoznaj się z sekcją Podstawowa konfiguracja.

Podstawowa konfiguracja

Te przewodniki wprowadzają podstawowe elementy konfiguracji Zabbixa:

- Logowanie i konfiguracja użytkownika
- Nowy host
- Nowa pozycja
- Nowy wyzwalacz
- Odbieranie powiadomień o problemach
- Nowy szablon

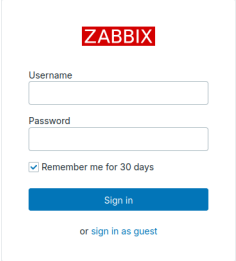
Logowanie i konfigurowanie użytkownika

Przegląd

W tej sekcji dowiesz się, jak zalogować się i skonfigurować użytkownika systemowego w Zabbix.

Logowanie

Otwórz adres URL frontendu Zabbix w przeglądarce: - Dla Apache: http://<server_ip_or_name>/zabbix - Dla NGINX: http://<server_ip_or_name



To jest ekran powitalny Zabbix. Wprowadź nazwę użytkownika **Admin** oraz hasło **zabbix**, aby zalogować się jako superużytkownik Zabbix. Uzyskasz dostęp do wszystkich sekcji menu.

Ze względów bezpieczeństwa zdecydowanie zaleca się natychmiastową zmianę domyślnego hasła konta Admin po pierwszym zalogowaniu.

Trwałe logowanie

Aby pozostać zalogowanym przez maksymalnie 30 dni, zaznacz opcję *Zapamiętaj na 30 dni* przed kliknięciem *Zaloguj się*.

Opcja *Zapamiętaj na 30 dni* włączona:

- Twoja sesja pozostaje aktywna przez 30 dni.
- *Automatyczne wylogowanie* zostaje zastąpione, dzięki czemu pozostajesz zalogowany do końca tego okresu.
- Przy kolejnych wizytach w ciągu 30 dni zostaniesz zalogowany automatycznie, bez ponownego wprowadzania danych uwierzytelniających.

Opcja *Zapamiętaj na 30 dni* wyłączona:

- Wcześniej włączone automatyczne logowanie zostaje wyczyszczone.
- Sesja wygaśnie zgodnie z skonfigurowanym interwałem *Automatyczne wylogowanie*.

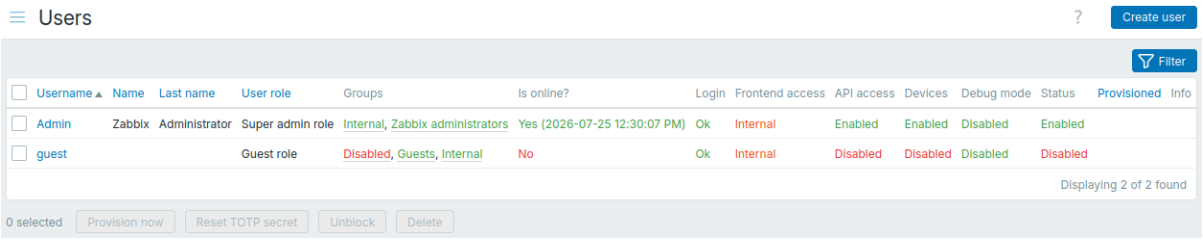
Ochrona przed atakami brute force

W przypadku pięciu kolejnych nieudanych prób logowania frontend Zabbix zostanie wstrzymany na 30 sekund, aby zapobiec atakom brute force i atakom słownikowym.

Adres IP nieudanej próby logowania zostanie wyświetlony po pomyślnym zalogowaniu.

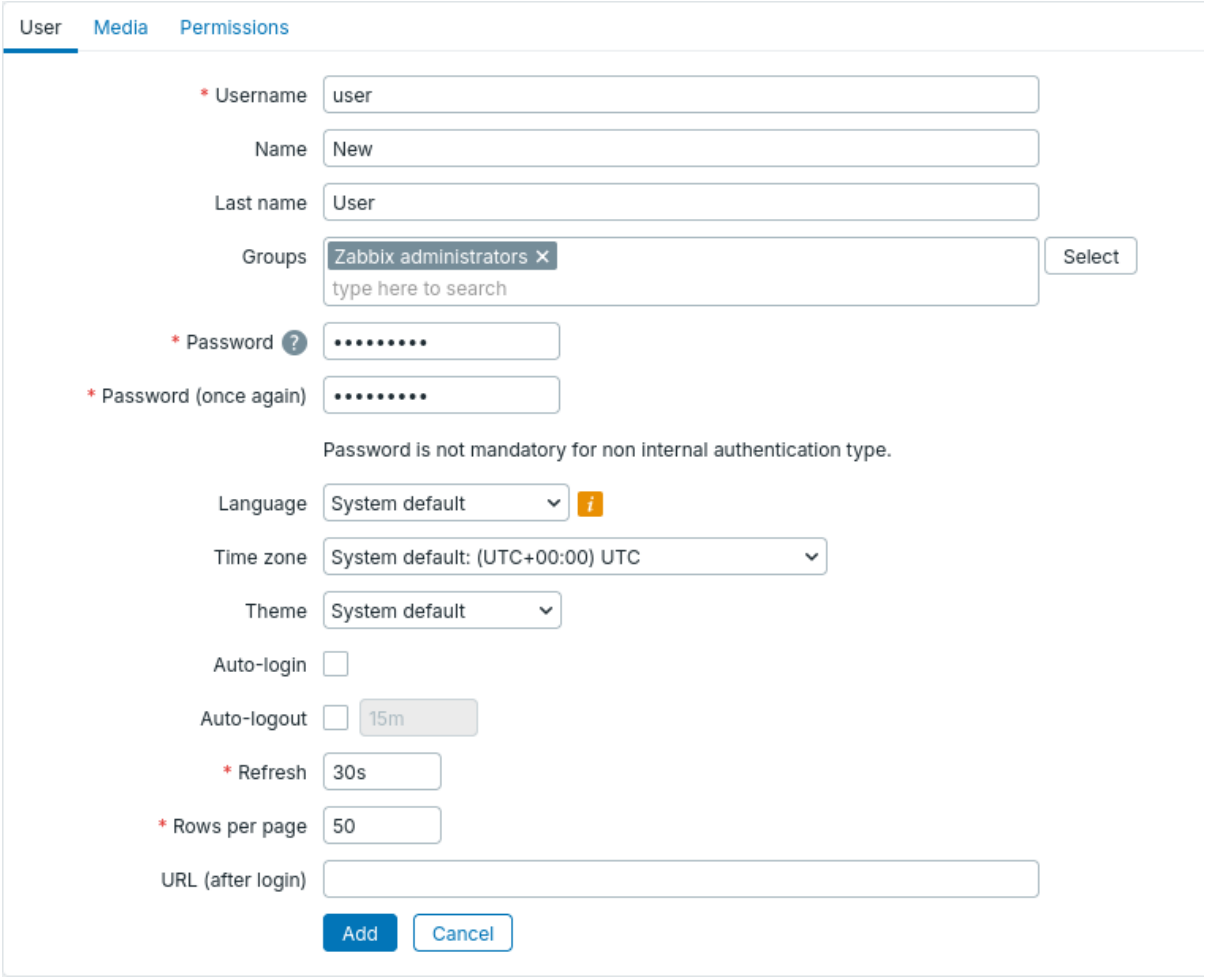
Dodawanie użytkownika

Aby wyświetlić informacje o użytkownikach, przejdź do *Users* > *Users* w pionowym menu bocznym.



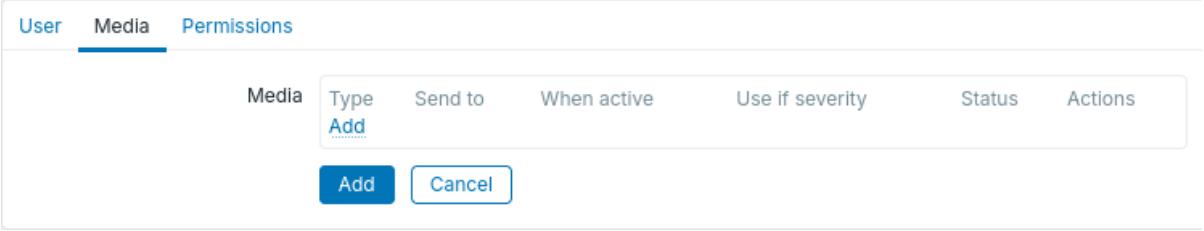
Aby dodać nowego użytkownika, wybierz *Create user* w prawym górnym rogu.

W formularzu nowego użytkownika upewnij się, że dodajesz użytkownika do jednej z istniejących grup użytkowników, na przykład 'Zabbix administrators'.



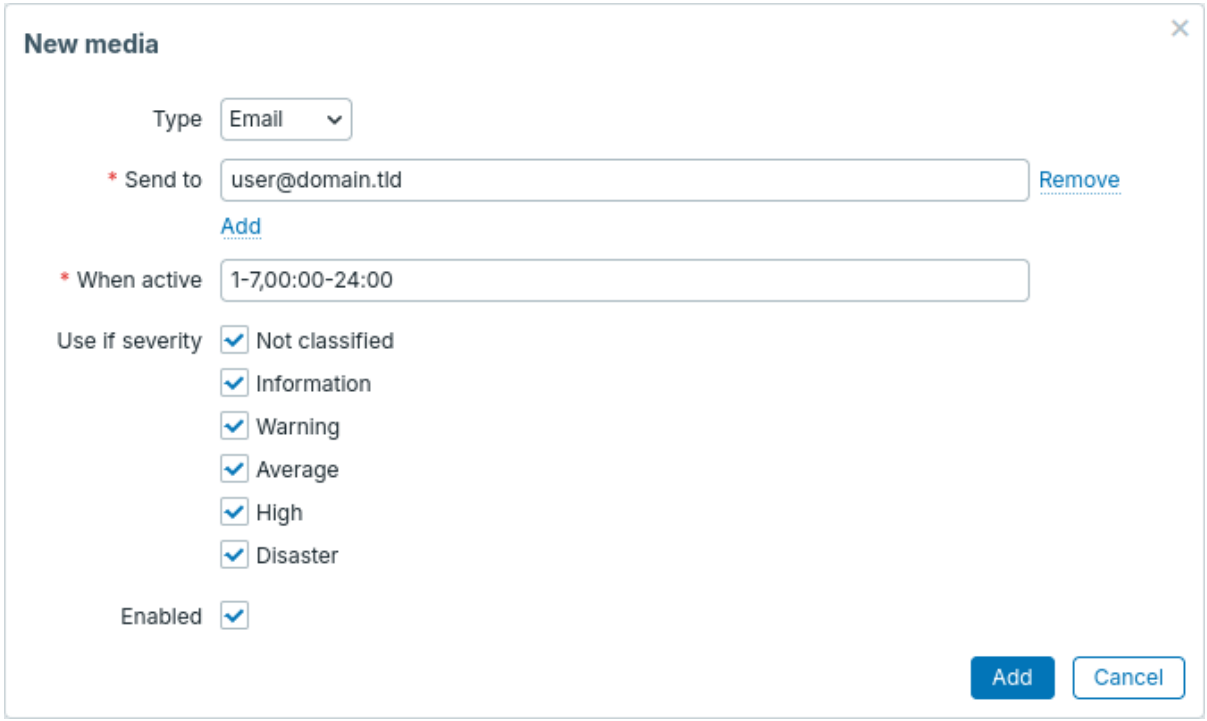
Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką. Szczegółowe informacje o polach wejściowych w tym formularzu konfiguracji znajdziesz na stronie User settings.

Domyślnie dla nowych użytkowników nie są zdefiniowane żadne media (metody dostarczania powiadomień). Aby utworzyć medium, przejdź do karty *Media* i kliknij *Add*.



W oknie podręcznym wprowadź adres e-mail użytkownika.
 Wszystkie typy mediów są domyślnie wyłączone i można je włączyć w *Alerts* > *Media types*.

Możesz określić okres czasu, w którym medium będzie aktywne (opis formatu znajduje się na stronie Time period specification). Domyślnie medium jest zawsze aktywne. Możesz też dostosować poziomy trigger severity, dla których medium będzie aktywne, ale na razie pozostaw wszystkie włączone.



Kliknij *Add*, aby zapisać medium, a następnie przejdź do karty *Permissions*.

Karta Permissions ma obowiązkowe pole *Role*. Rola określa, które elementy frontend użytkownik może wyświetlać i jakie działania może wykonywać. Kliknij *Select* i wybierz jedną z ról z listy. Na przykład wybierz *Admin role*, aby umożliwić dostęp do wszystkich sekcji frontend Zabbixa, z wyjątkiem *Administration*. Później możesz zmodyfikować uprawnienia lub utworzyć więcej ról użytkowników. Po wybraniu roli uprawnienia pojawią się na tej samej karcie:

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

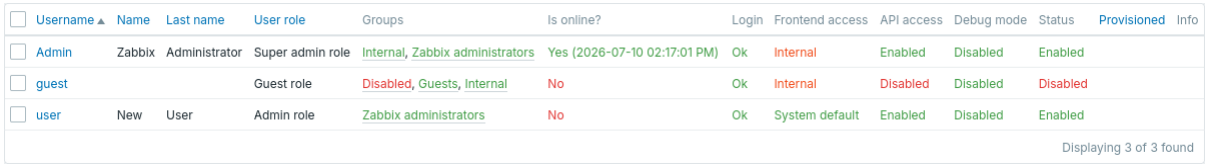
Enabled | Manage own devices

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

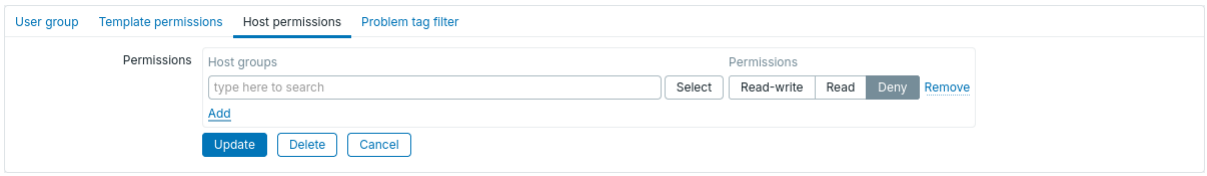
Add | Cancel

Kliknij *Add* w formularzu właściwości użytkownika, aby zapisać użytkownika. Nowy użytkownik pojawi się na liście użytkowników.

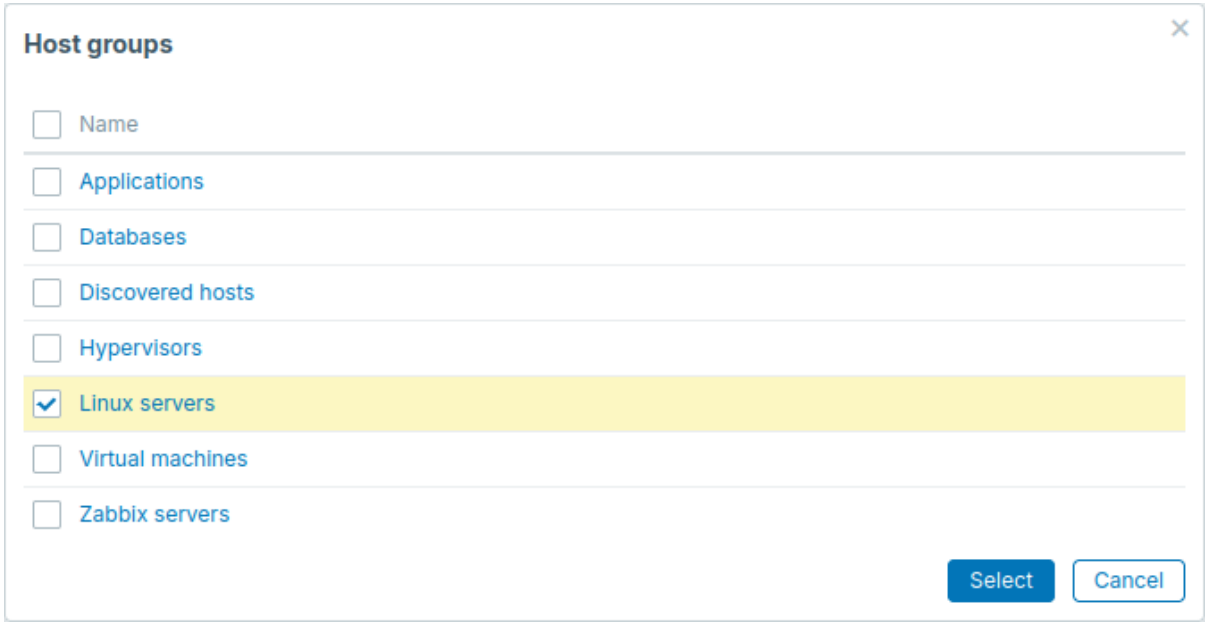


Dodawanie uprawnień

Domyślnie nowy użytkownik nie ma uprawnień dostępu do hostów i szablonów. Aby nadać użytkownikowi uprawnienia, kliknij grupę użytkownika w kolumnie *Groups* (w tym przypadku - *Zabbix administrators*). W formularzu właściwości *User groups* przejdź do karty *Host permissions*, aby przypisać uprawnienia do grup hostów. Kliknij Add, aby wyświetlić pole wyboru grupy hostów:



Następnie kliknij *Select* obok pola, aby zobaczyć listę grup hostów. Ten użytkownik ma mieć dostęp tylko do odczytu do grupy *Linux servers*, więc zaznacz odpowiednie pole wyboru na liście i kliknij *Select*, aby potwierdzić wybór.



Kliknij przycisk *Read*, aby ustawić poziom uprawnień, a następnie *Update*, aby zapisać zmiany w konfiguracji grupy użytkowników.

Aby nadać uprawnienia do szablonów, należy przejść do karty *Template permissions* i określić grupy szablonów. Kroki są identyczne jak podczas przypisywania uprawnień do grup hostów. Przegląd szablonów jest dostępny w sekcji New template tego przewodnika Quickstart.

Attention:
W Zabbix prawa dostępu do hostów i szablonów są przypisywane do grup użytkowników, a nie do poszczególnych użytkowników.

Gotowe! Możesz spróbować zalogować się przy użyciu danych uwierzytelniających nowego użytkownika.

Nowy host

Przegląd

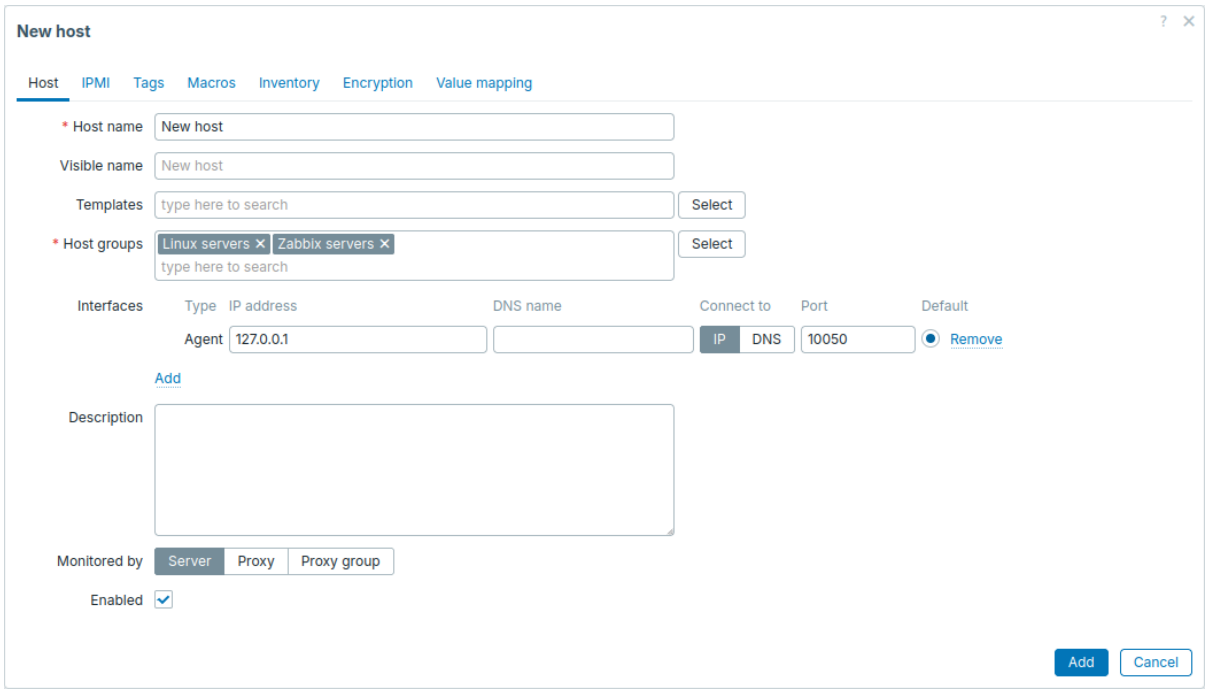
W tej sekcji dowiesz się, jak skonfigurować nowy host.

Host w Zabbix to obiekt sieciowy (fizyczny, wirtualny), który chcesz monitorować. Definicja tego, czym może być ”host” w Zabbix, jest dość elastyczna. Może to być fizyczny serwer, przełącznik sieciowy, maszyna wirtualna lub jakaś aplikacja.

Dodawanie hosta

Informacje o skonfigurowanych hostach w Zabbix są dostępne w sekcji menu *Zbieranie danych* > *Hosty*, a także w sekcji menu *Monitorowanie* > *Hosty*. Istnieje już jeden wstępnie zdefiniowany host o nazwie „Zabbix server", ale chcemy nauczyć się dodawać kolejny.

Aby dodać nowy host, kliknij *Utwórz hosta*. Zostanie otwarty formularz konfiguracji hosta.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Minimalny zakres informacji, który należy tutaj wprowadzić, obejmuje:

Nazwa hosta

- Wprowadź nazwę hosta. Dozwolone są znaki alfanumeryczne, spacje, kropki, myślniki i podkreślenia.

Grupy hostów

- Wybierz jedną lub kilka istniejących grup, klikając przycisk *Wybierz*, albo wprowadź nazwę nieistniejącej grupy, aby utworzyć nową grupę.

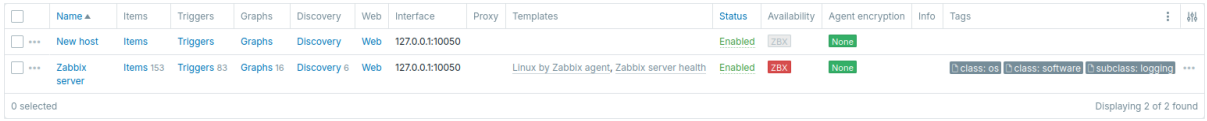
Note:
Wszystkie uprawnienia dostępu są przypisywane do grup hostów, a nie do poszczególnych hostów. Dlatego host musi należeć co najmniej do jednej grupy.

Interfejsy: adres IP

- Chociaż technicznie nie jest to pole wymagane, interfejs hosta jest niezbędny do zbierania określonych metryk. Aby używać pasywnych kontroli agenta Zabbix, określ w tym polu adres IP lub DNS agenta. Pamiętaj, że należy również określić adres IP lub DNS serwera Zabbix w dyrektywie „Server" w pliku konfiguracyjnym agenta Zabbix. Jeśli agent Zabbix i serwer Zabbix są zainstalowane na tej samej maszynie, należy określić ten sam adres IP/DNS w obu miejscach.

Pozostałe opcje można pozostawić bez zmian, korzystając z ich wartości domyślnych.

Po zakończeniu kliknij *Dodaj*. Nowy host powinien być widoczny na liście hostów.



Kolumna *Dostępność* zawiera wskaźniki dostępności hosta dla każdego interfejsu. Po zdefiniowaniu interfejsu agenta Zabbix możemy użyć ikony dostępności agenta (oznaczonej jako *ZBX*), aby określić dostępność hosta:

- ZBX - status hosta nie został ustalony; nie wykonano jeszcze żadnej kontroli metryki.
- ZBX - host jest dostępny, kontrola metryki zakończyła się powodzeniem.
- ZBX - host jest niedostępny, kontrola metryki zakończyła się niepowodzeniem (najedź kursorem myszy na ikonę, aby wyświetlić komunikat o błędzie). Mógł wystąpić błąd komunikacji, prawdopodobnie spowodowany nieprawidłowymi danymi uwierzytelniającymi interfejsu. Sprawdź, czy serwer Zabbix działa, i spróbuj również odświeżyć stronę później.

Nowa pozycja

Przegląd

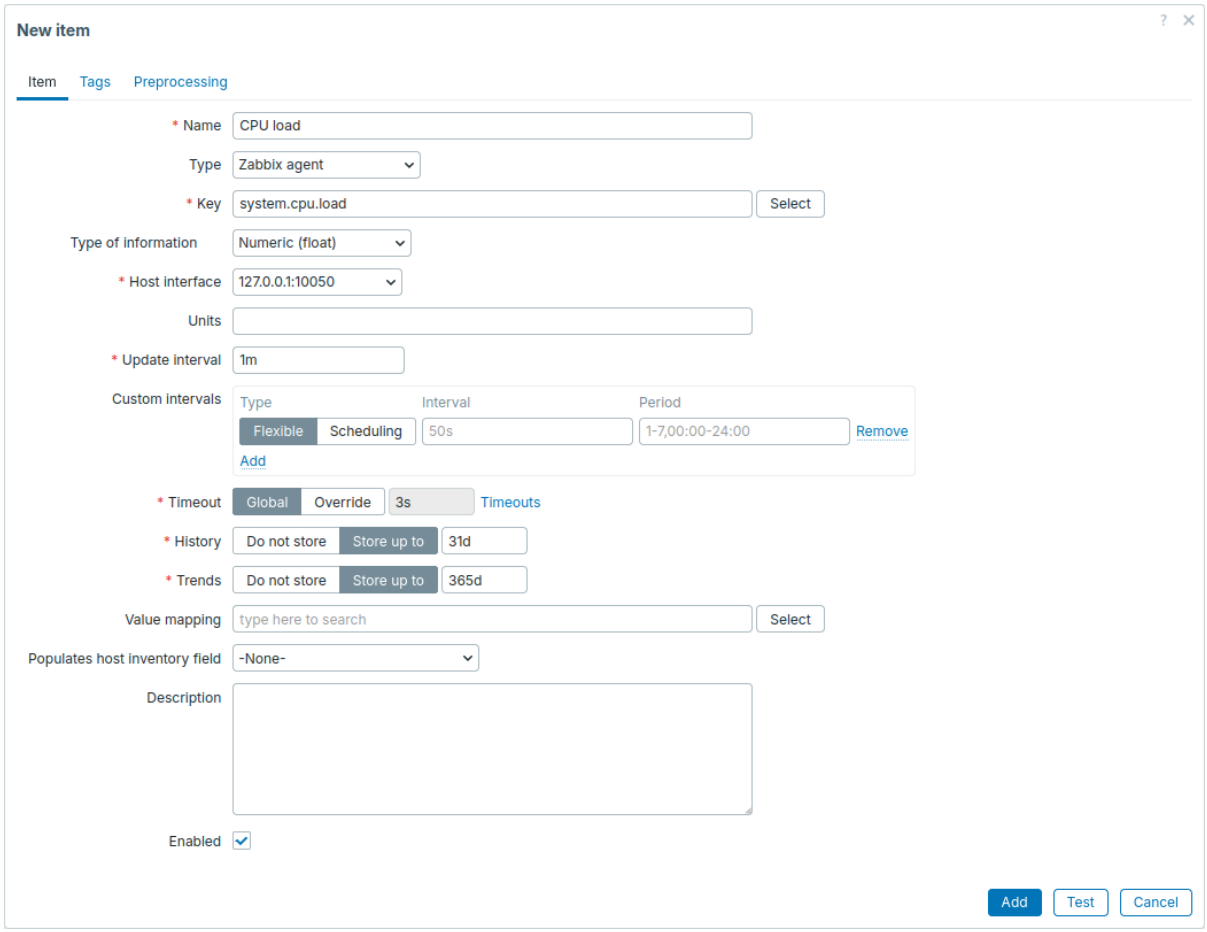
W tej sekcji dowiesz się, jak skonfigurować pozycję.

Pozycje są podstawą zbierania danych w Zabbix. Bez pozycji nie ma danych - ponieważ tylko pozycja definiuje pojedynczą metrykę lub rodzaj danych do zebrania z hosta.

Dodawanie pozycji

Wszystkie pozycje są grupowane wokół hostów. Dlatego, aby skonfigurować przykładową pozycję, przechodzimy do *Data collection* > *Hosts* i znajdujemy utworzony przez nas "New host".

Kliknij link *Items* w wierszu "New host", a następnie kliknij *Create item*. Zostanie otwarty formularz konfiguracji pozycji.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W przypadku naszej przykładowej pozycji podstawowe informacje do wprowadzenia to:

Name

- Wprowadź wartość *CPU load*. Będzie to nazwa pozycji wyświetlana na listach i w innych miejscach.

Key

- Ręcznie wprowadź wartość *system.cpu.load*. Jest to techniczna nazwa pozycji, która identyfikuje typ informacji, jakie będą zbierane. Ten konkretny klucz to tylko jeden z pre-defined keys dostarczanych z agentem Zabbix.

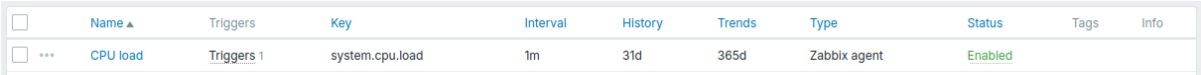
Type of information

- Ten atrybut definiuje format oczekiwanych danych. Dla klucza *system.cpu.load* to pole zostanie automatycznie ustawione na *Numeric (float)*.

Note:
Możesz również zmniejszyć liczbę dni, przez które będzie przechowywana item history, do 7 lub 14. To dobra praktyka, ponieważ odciąża bazę danych od przechowywania dużej liczby wartości historycznych.

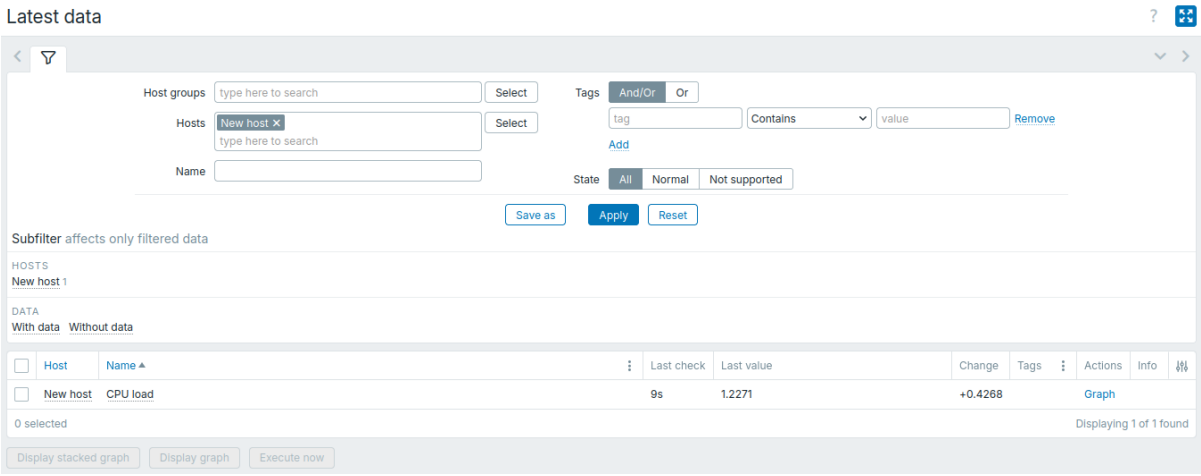
Other options można pozostawić bez zmian, używając wartości domyślnych.

Po zakończeniu kliknij *Add*. Nowa pozycja powinna pojawić się na liście pozycji.



Wyświetlanie danych

Po zdefiniowaniu pozycji możesz chcieć sprawdzić, czy rzeczywiście zbiera ona dane. W tym celu przejdź do *Monitoring* > *Latest data*, wybierz 'New host' w filtrze i kliknij *Apply*.



Należy jednak pamiętać, że pojawienie się pierwszych danych może potrwać do 60 sekund. Domyślnie z taką częstotliwością serwer odczytuje zmiany konfiguracji i wykrywa nowe pozycje do wykonania.

Jeśli w kolumnie 'Change' nie widzisz wartości, możliwe, że do tej pory odebrano tylko jedną wartość. Poczekaj 30 sekund na nadejście kolejnej wartości.

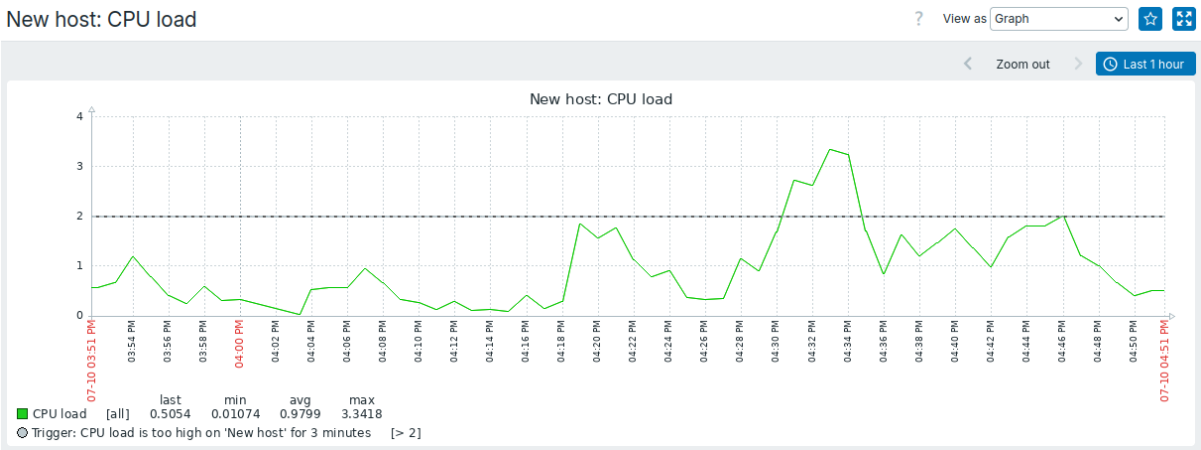
Jeśli nie widzisz informacji o pozycji jak na zrzucie ekranu, upewnij się, że:

- pola pozycji 'Key' i 'Type of information' zostały wypełnione dokładnie tak jak na zrzucie ekranu;
- zarówno agent, jak i serwer są uruchomione;
- status hosta to 'Monitored', a jego ikona dostępności jest zielona;
- host wybrany w filtrze hosta jest prawidłowy;
- pozycja jest włączona.

Wykresy

Gdy pozycja działa już przez jakiś czas, warto zobaczyć coś wizualnego. Proste wykresy są dostępne dla każdej monitorowanej pozycji numerycznej bez dodatkowej konfiguracji. Wykresy te są generowane w czasie wykonywania.

Aby wyświetlić wykres, przejdź do *Monitoring* > *Latest data* i kliknij łącze *Graph* obok pozycji.



Nowy wyzwalacz

Przegląd

W tej sekcji dowiesz się, jak skonfigurować wyzwalacz.

Pozycje tylko zbierają dane. Aby automatycznie oceniać przychodzące dane, musimy zdefiniować wyzwalacze. Wyzwalacz zawiera wyrażenie, które definiuje próg określający, jaki poziom danych jest akceptowalny.

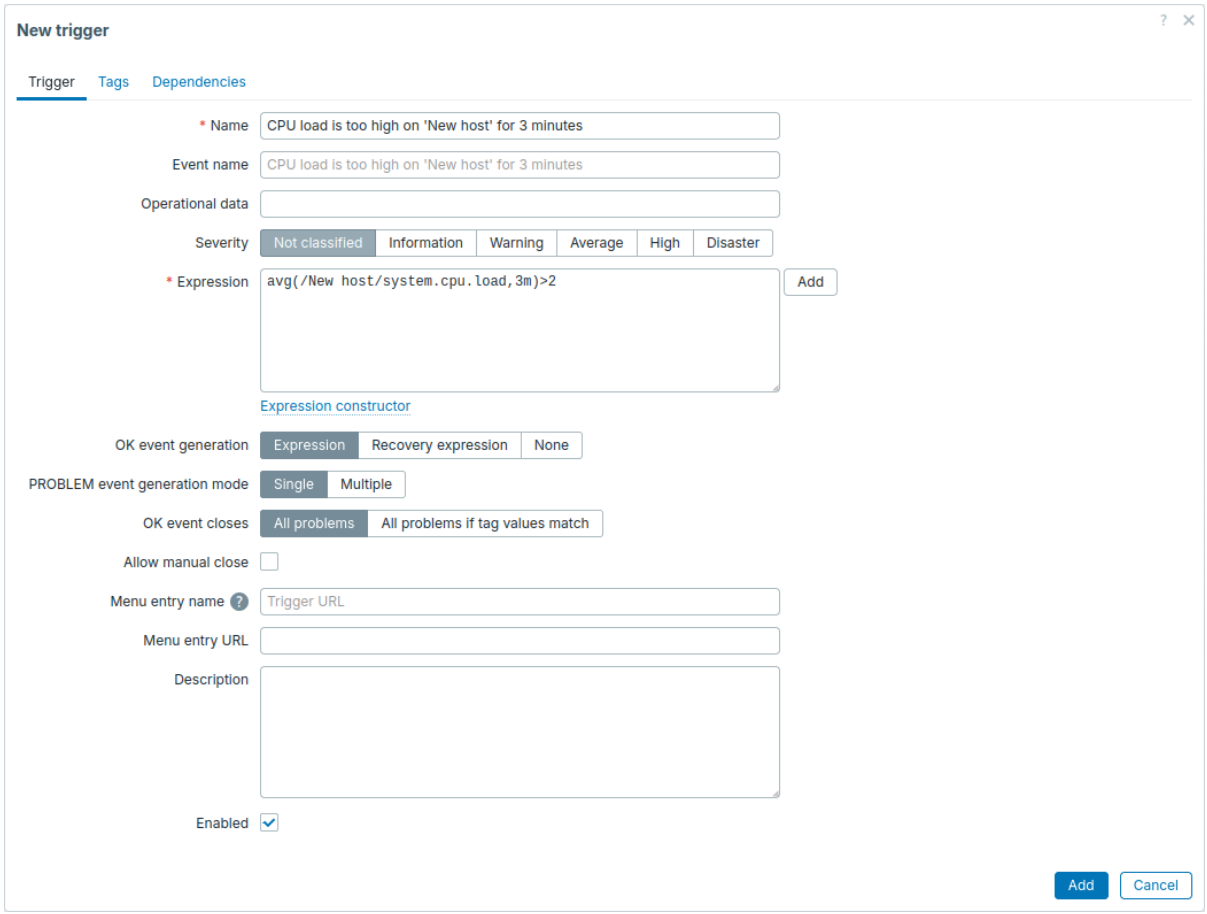
Jeśli ten poziom zostanie przekroczony przez przychodzące dane, wyzwalacz zostanie „uruchomiony" lub przejdzie w stan „Problem" - co oznacza, że wydarzyło się coś, co może wymagać uwagi. Jeśli poziom ponownie stanie się akceptowalny, wyzwalacz wraca do stanu „Ok".

Dodawanie wyzwalacza

Aby skonfigurować wyzwalacz dla naszej pozycji:

1. Przejdź do *Zbieranie danych* > *Hosty*.
2. Kliknij link *Wyzwalacze* w wierszu „New host".
3. Kliknij *Utwórz wyzwalacz*.

Zostanie otwarty formularz konfiguracji wyzwalacza.



W przypadku naszego wyzwalacza najważniejsze informacje do wprowadzenia tutaj to:

Name

- Wprowadź wartość *CPU load is too high on 'New host' for 3 minutes*. Będzie to nazwa wyzwalacza wyświetlana na listach i w innych miejscach.

Expression

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

To jest wyrażenie wyzwalacza. Upewnij się, że wyrażenie zostało wpisane poprawnie, aż do ostatniego znaku. Klucz pozycji tutaj (`system.cpu.load`) służy do odwołania się do pozycji. To konkretne wyrażenie zasadniczo oznacza, że próg problemu zostaje przekroczony, gdy średnia wartość obciążenia CPU z 3 minut jest większa niż 2. Więcej informacji znajdziesz w składni wyrażeń wyzwalaczy.

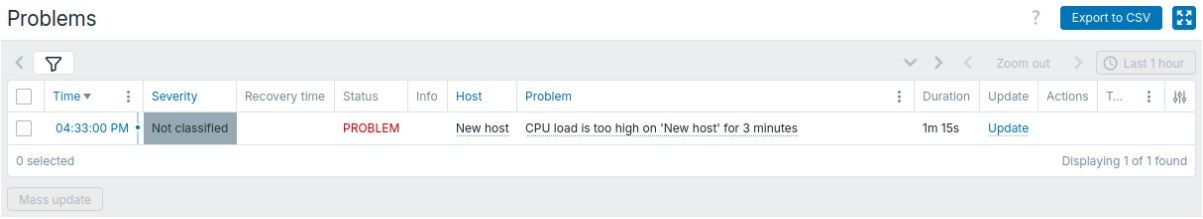
Pozostałe opcje można pozostawić bez zmian, używając ich wartości domyślnych.

Po zakończeniu kliknij *Dodaj*. Nowy wyzwalacz powinien pojawić się na liście wyzwalaczy.

Wyświetlanie stanu wyzwalacza

Po zdefiniowaniu wyzwalacza możesz chcieć sprawdzić jego stan.

Jeśli obciążenie CPU przekroczyło poziom progowy zdefiniowany w wyzwalaczu, problem zostanie wyświetlony w *Monitoring* > *Problems*.



Miganie w kolumnie statusu wskazuje na niedawną zmianę stanu wyzwalacza, która nastąpiła w ciągu ostatnich 30 minut.

Otrzymywanie powiadomienia o problemie

Przegląd

W tej sekcji dowiesz się, jak skonfigurować alertowanie w postaci powiadomień w Zabbix.

Ponieważ pozycje zbierają dane, a wyzwalacze są zaprojektowane tak, aby „uruchamiać się" w sytuacjach problemowych, przydatne byłoby również wdrożenie mechanizmu alertowania, który powiadamiałby o ważnych zdarzeniach nawet wtedy, gdy frontend Zabbix nie jest aktywnie sprawdzany.

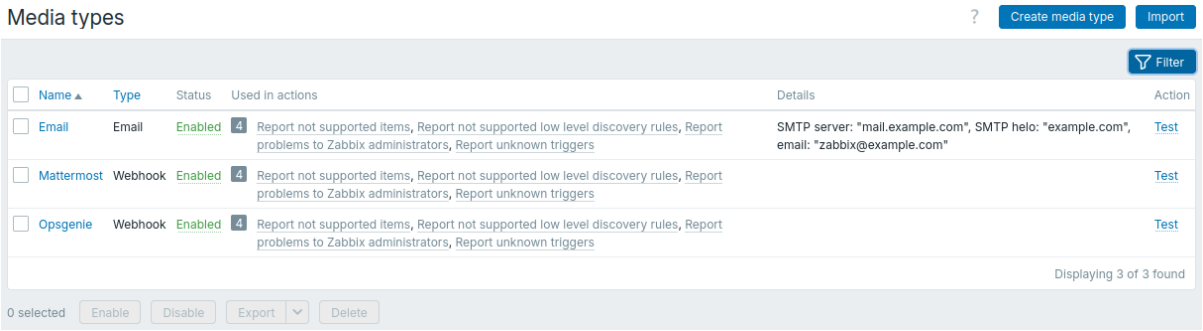
Właśnie to robią powiadomienia.

Ponieważ poczta e-mail jest najpopularniejszą metodą dostarczania powiadomień o problemach, nauczymy się, jak skonfigurować powiadomienie e-mail.

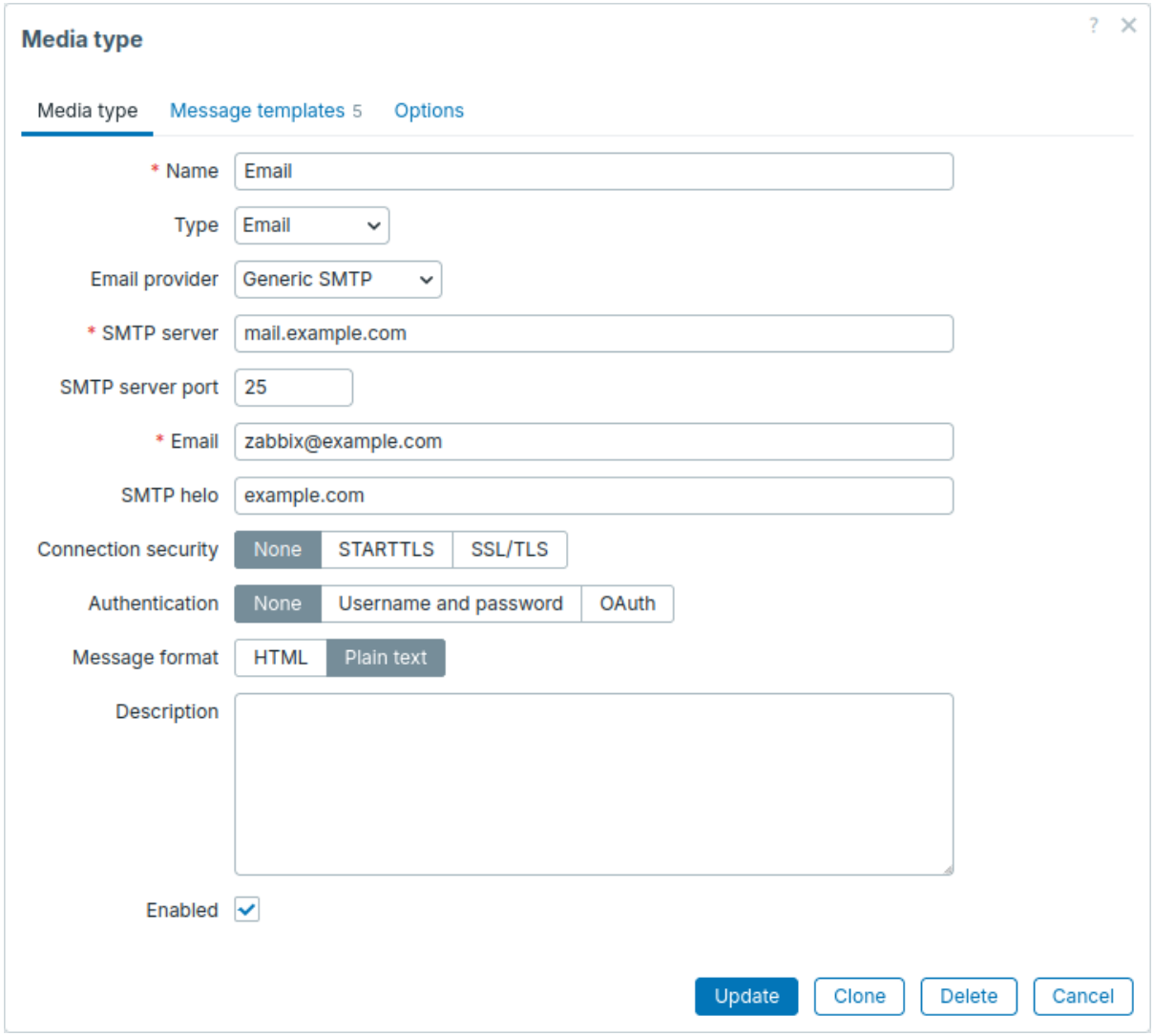
Ustawienia e-mail

Początkowo w Zabbix dostępnych jest kilka predefiniowanych metod dostarczania powiadomień. Email jest jedną z nich.

Aby skonfigurować ustawienia e-mail, przejdź do *Alerts* > *Media types* i kliknij *Email* na liście predefiniowanych typów mediów.



Zostanie otwarty formularz konfiguracji ustawień e-mail.



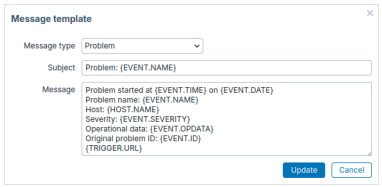
Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Na karcie *Media type* ustaw wartości *SMTP server*, *SMTP helo* oraz *Email* odpowiednio do swojego środowiska.

Note:
Wartość w polu *Email* będzie używana jako adres „From” dla powiadomień wysyłanych z Zabbix.

Następnie należy zdefiniować treść komunikatu o problemie. Treść jest definiowana za pomocą szablonu wiadomości, skonfigurowanego na karcie *Message templates*.

Kliknij *Add*, aby utworzyć szablon wiadomości, i wybierz *Problem* jako typ wiadomości.

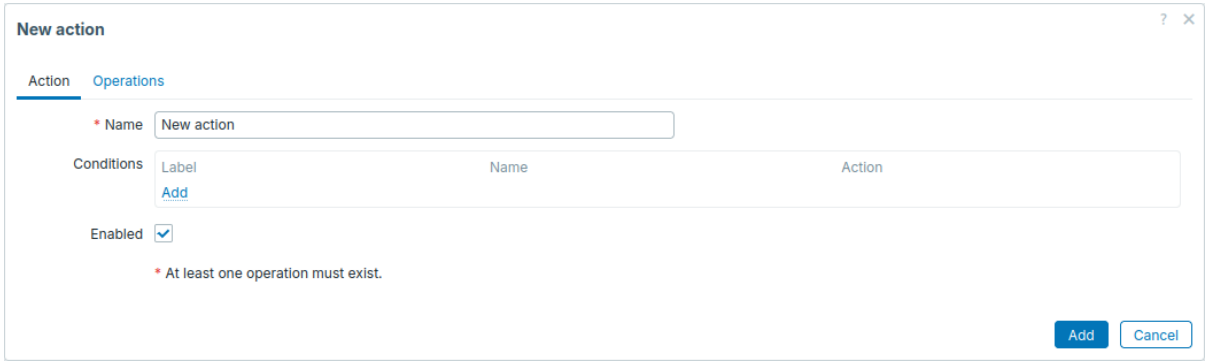


Gdy wszystko będzie gotowe, kliknij *Add* i zapisz formularz.

Teraz skonfigurowałeś *Email* jako działający typ mediów. Typ mediów musi być również powiązany z użytkownikami poprzez zdefiniowanie konkretnych adresów dostarczania (tak jak zrobiliśmy to podczas konfigurowania nowego użytkownika); w przeciwnym razie nie będzie używany.

Nowa akcja

Dostarczanie powiadomień jest jedną z rzeczy, które w Zabbix wykonują actions. Dlatego, aby skonfigurować powiadomienie, przejdź do *Alerts* > *Actions* > *Trigger actions* i kliknij *Create action*.

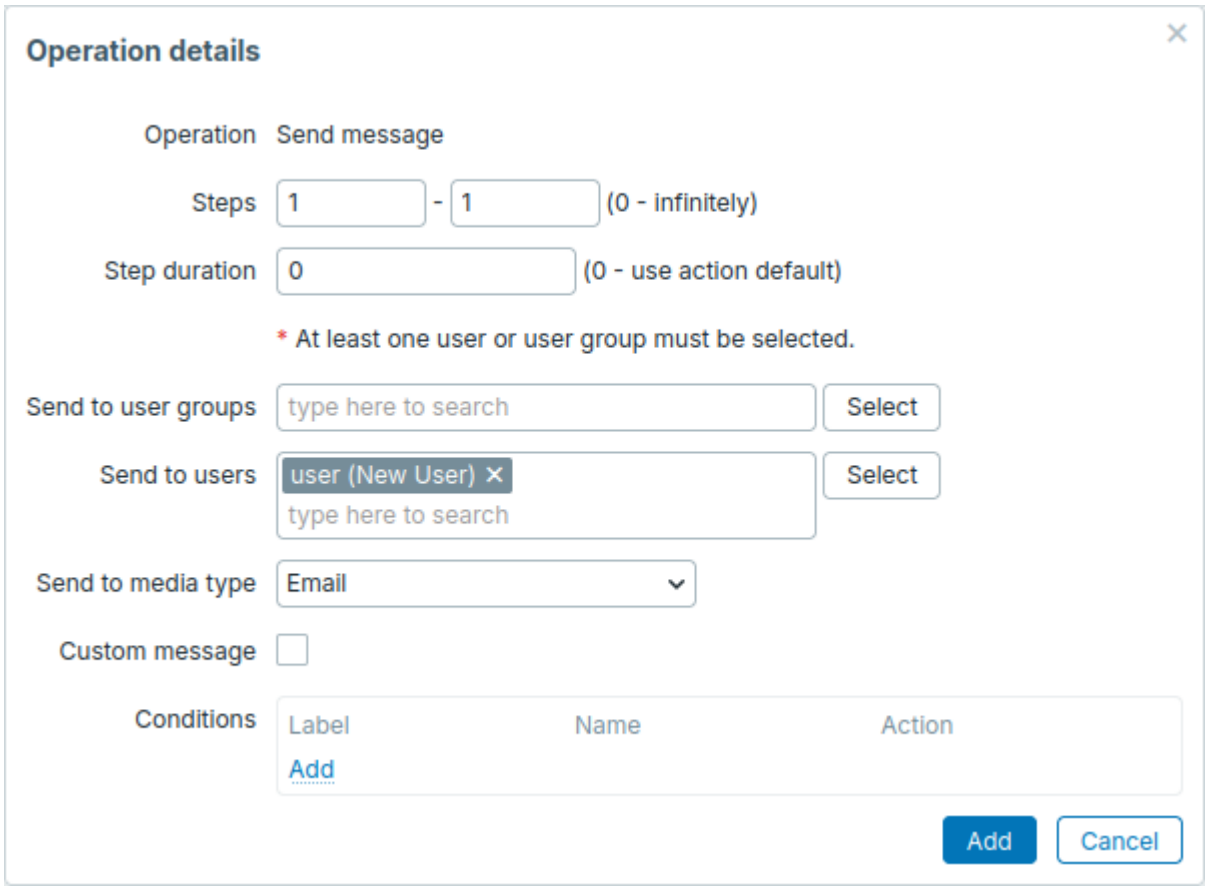


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W tym formularzu wprowadź nazwę akcji.

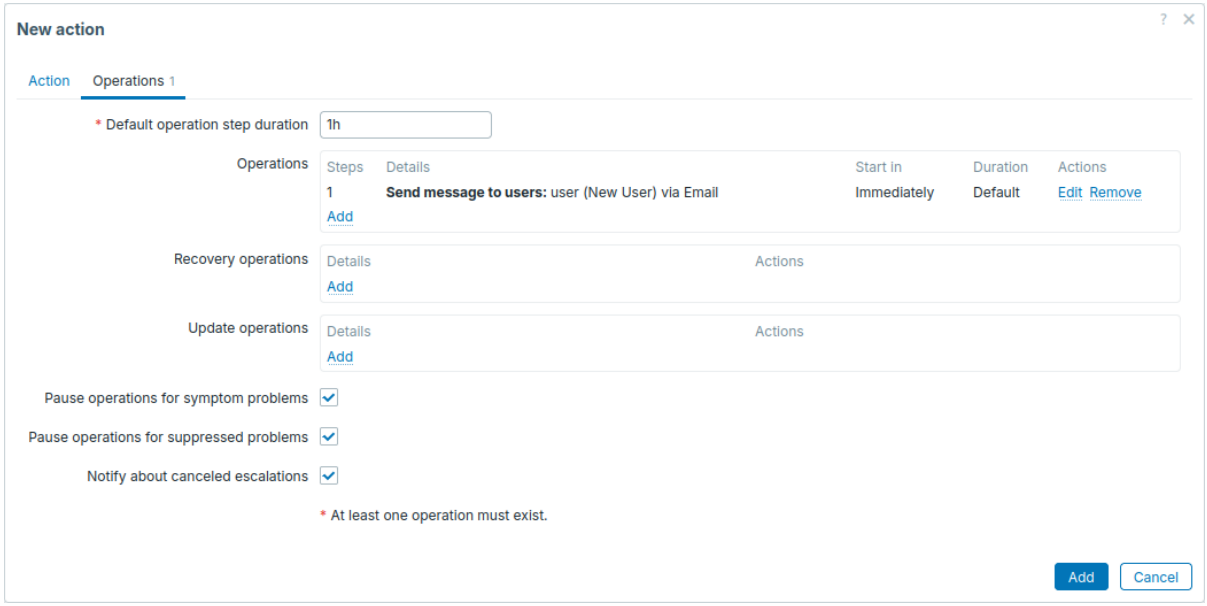
W najprostszym przypadku, jeśli nie dodamy żadnych bardziej szczegółowych conditions, akcja zostanie wykonana przy każdej zmianie wyzwalacza z 'Ok' na 'Problem'.

Nadal musimy określić, co akcja ma robić - i robi się to na karcie *Operations*. Kliknij *Add* w sekcji *Operations*, co otworzy nowy formularz operacji.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Tutaj kliknij *Select* w sekcji *Send to Users* i wybierz użytkownika ('user'), którego zdefiniowaliśmy. Wybierz "Email" jako wartość *Send to media type*. Po wykonaniu tego kliknij *Add*, a operacja powinna zostać dodana:



To wszystko w przypadku prostej konfiguracji akcji, więc kliknij *Add* w formularzu akcji.

Odbieranie powiadomienia

Teraz, gdy dostarczanie powiadomień jest skonfigurowane, warto faktycznie otrzymać jedno z nich. Aby to ułatwić, możemy celowo zwiększyć obciążenie na naszym host - tak, aby nasz wyzwalacz „zadziałał" i abyśmy otrzymali powiadomienie o problemie.

Otwórz konsolę na swoim host i uruchom:

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

Możesz uruchomić jeden lub kilka z tych procesów.

Następnie przejdź do *Monitoring* > *Latest data* i sprawdź, jak wzrosły wartości „CPU Load". Pamiętaj, aby nasz wyzwalacz „zadziałał", wartość „CPU Load" musi przekraczać „2" przez 3 minuty z rzędu. Gdy to nastąpi:

- w *Monitoring* > *Problems* powinieneś zobaczyć wyzwalacz z migającym stanem `Problem`;
- powinieneś otrzymać powiadomienie o problemie w swojej poczcie e-mail.

Attention:
Jeśli powiadomienia nie działają:

- sprawdź ponownie, czy zarówno ustawienia poczty e-mail, jak i akcja zostały poprawnie skonfigurowane
- upewnij się, że utworzony użytkownik ma co najmniej uprawnienia do odczytu na host, który wygenerował zdarzenie, jak opisano w kroku *Adding user*. Użytkownik, będąc częścią grupy użytkowników „Zabbix administrators", musi mieć co najmniej dostęp do odczytu do grupy hostów „Linux servers", do której należy nasz host.
- Dodatkowo możesz sprawdzić dziennik akcji, przechodząc do *Reports* > *Action log*.

Nowy szablon

Omówienie

W tej sekcji dowiesz się, jak skonfigurować szablon.

Wcześniej dowiedzieliśmy się, jak skonfigurować pozycję i wyzwalacz oraz jak uzyskać powiadomienie o problemie dotyczącym hosta.

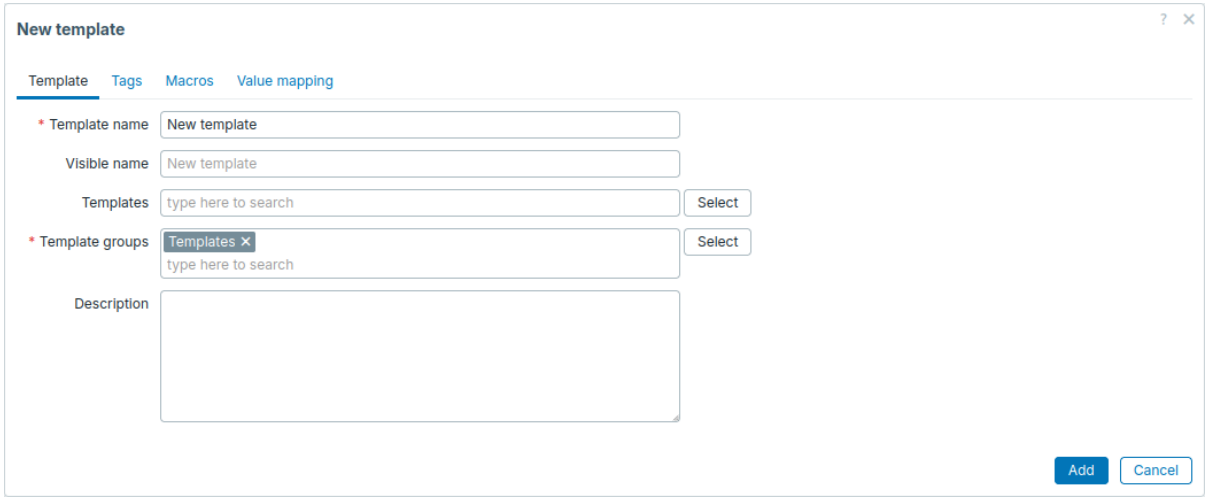
Chociaż wszystkie te kroki same w sobie zapewniają dużą elastyczność, wykonanie ich dla, powiedzmy, tysiąca hostów może wydawać się czasochłonne. Przydatna byłaby pewna automatyzacja.

Właśnie tutaj z pomocą przychodzą szablony. Szablony umożliwiają grupowanie przydatnych pozycji, wyzwalaczy i innych encji, dzięki czemu można ich wielokrotnie używać, stosując je do hostów w jednym kroku.

Po połączeniu szablonu z hostem host dziedziczy wszystkie encje szablonu. Dzięki temu można bardzo szybko zastosować gotowy zestaw kontroli.

Dodawanie szablonu

Aby rozpocząć pracę z szablonami, najpierw musimy utworzyć jeden z nich. W tym celu w sekcji *Data collection* > *Templates* kliknij *Create template*. Spowoduje to wyświetlenie formularza konfiguracji szablonu.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Wymagane parametry do wprowadzenia tutaj to:

Template name

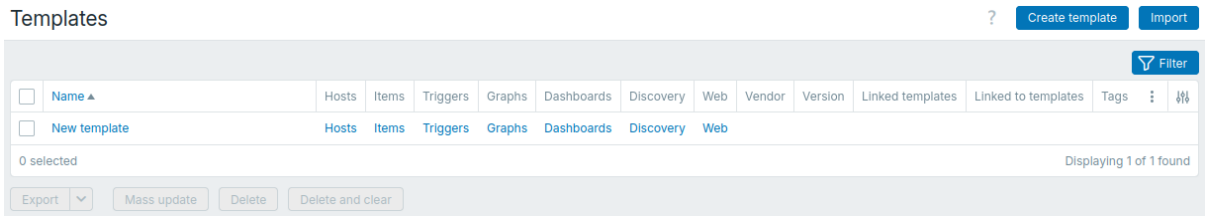
- Wprowadź nazwę szablonu. Dozwolone są znaki alfanumeryczne, spacje i podkreślenia.

Template groups

- Wybierz jedną lub kilka grup, klikając przycisk *Select*. Szablon musi należeć do grupy.

Note:
Uprawnienia dostępu do grup szablonów są przypisywane w konfiguracji user group na karcie **Template permissions** w taki sam sposób jak uprawnienia hostów. Wszystkie uprawnienia dostępu są przypisywane do grup, a nie do pojedynczych szablonów, dlatego dodanie szablonu do co najmniej jednej grupy jest obowiązkowe.

Po zakończeniu kliknij *Add*. Twój nowy szablon powinien być widoczny na liście szablonów. Możesz również użyć filtra, aby znaleźć swój szablon.

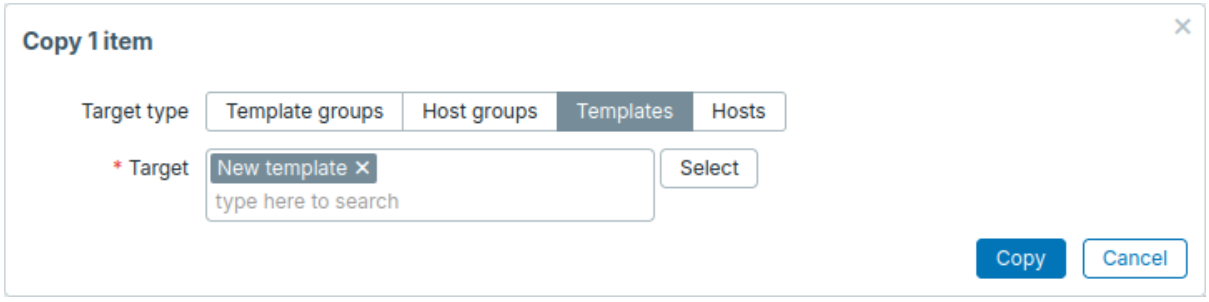


Jak widać, szablon już tam jest, ale nie zawiera niczego - żadnych pozycji, wyzwalaczy ani innych encji.

Dodawanie pozycji do szablonu

Aby dodać pozycję do szablonu:

1. Przejdź do *Gromadzenie danych* > *Hosty*.
2. Znajdź na liście „New host” i kliknij hiperłącze *Pozycje*.
3. Zaznacz pole wyboru pozycji „CPU Load” na liście.
4. Kliknij przycisk *Kopiuj* poniżej listy.
5. Wybierz kartę *Szablony*.
6. Wybierz szablon, do którego chcesz skopiować pozycję.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką. 7. Kliknij przycisk *Kopiuj*.

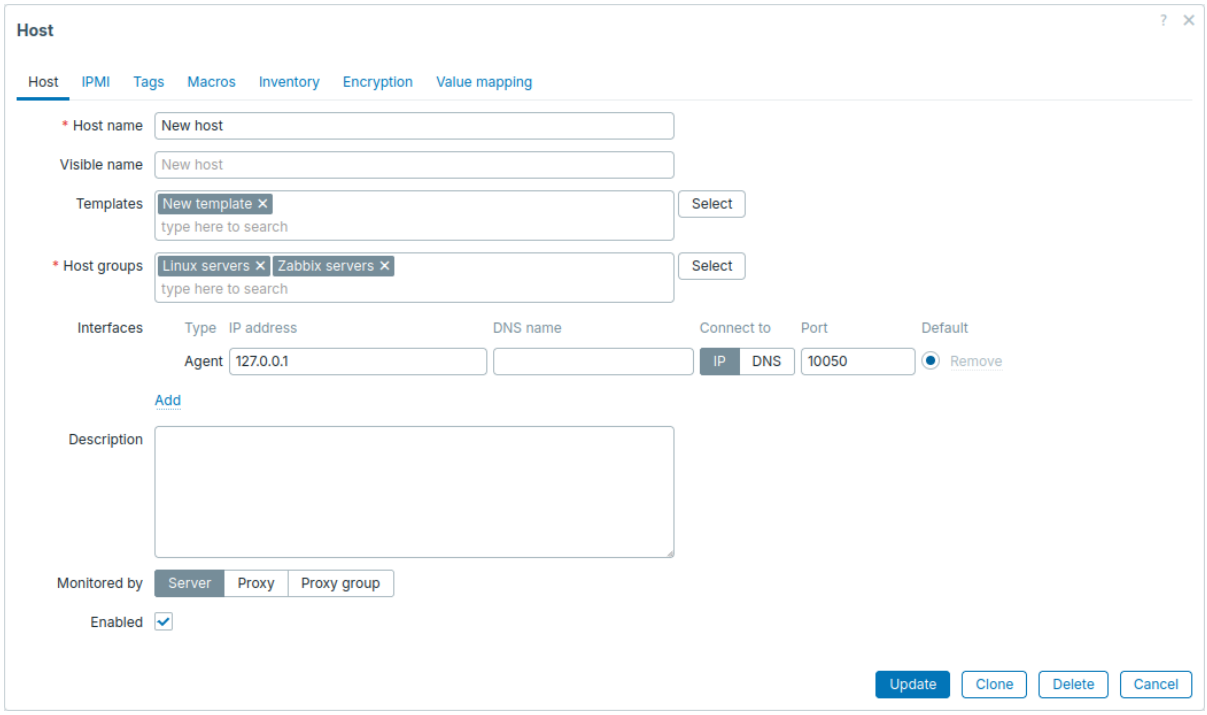
Jeśli teraz przejdziesz do *Gromadzenie danych* > *Szablony*, szablon „New template” powinien zawierać jedną nową pozycję.

Na razie zakończymy na jednej pozycji, ale w podobny sposób możesz dodawać do szablonu dowolne inne pozycje, wyzwalacze lub inne encje, aż uzyskasz dość kompletny zestaw encji dla danego celu (monitorowania systemu operacyjnego, monitorowania pojedynczej aplikacji).

Łączenie szablonu z hostem

Po przygotowaniu szablonu pozostaje tylko dodać go do hosta. W tym celu:

1. Przejdź do *Zbieranie danych* > *Hosty*.
2. Kliknij „Nowy host”, aby otworzyć formularz konfiguracji, i znajdź pole *Szablony*.
3. Zacznij wpisywać *Nowy szablon* w polu *Szablony*. Nazwa nowo utworzonego szablonu powinna pojawić się na liście rozwijanej.
4. Przewiń listę i wybierz szablon. Sprawdź, czy pojawił się w polu *Szablony*.



5. Kliknij *Aktualizuj* w formularzu, aby zapisać zmiany.

Szablon został teraz dodany do hosta wraz ze wszystkimi zawartymi w nim encjami.

W ten sposób można zastosować go również do dowolnego innego hosta. Wszelkie zmiany dotyczące pozycji, wyzwalaczy i innych encji na poziomie szablonu zostaną przekazane do hostów, z którymi szablon jest połączony.

Łączenie predefiniowanych szablonów z hostami

Jak mogłeś zauważyć, Zabbix zawiera zestaw predefiniowanych szablonów dla różnych systemów operacyjnych, urządzeń i aplikacji. Aby bardzo szybko rozpocząć monitorowanie, możesz przypisać odpowiedni z nich do hosta, ale pamiętaj, że te szablony wymagają dostosowania do Twojego środowiska. Niektóre kontrole mogą nie być potrzebne, a interwały odpytywania mogą być zbyt częste.

Dostępnych jest więcej informacji o szablonach.

Monitorowanie serwera WWW Apache

Aby rozpocząć monitorowanie serwera WWW Apache, wykonaj następujące kroki:

1. Upewnij się, że masz dostęp do statystyk Apache.
2. Zastosuj szablon monitorowania Apache.
3. Sprawdź najnowsze dane.

Aby skorzystać z tego przewodnika, musisz mieć zainstalowany serwer Zabbix oraz frontend installed. Musisz być zalogowany do Zabbix jako domyślny użytkownik *Admin*.

Dostęp do statystyk Apache

Celem tego kroku jest upewnienie się, że statystyki serwera WWW Apache mogą być odczytywane przez Zabbix.

W Apache informacje w czasie rzeczywistym o wydajności i aktywności serwera są udostępniane przez wbudowany moduł `mod_status`. Po włączeniu tworzy on stronę dostępną przez WWW (zwykle pod adresem `/server-status`), która pokazuje dane o wydajności i aktywności.

Aby sprawdzić, czy moduł `mod_status` jest włączony dla serwera WWW Apache, uruchom:

Na Debian/Ubuntu	Na systemach opartych na RHEL
`apache2ctl -M | grep status`	`httpd -M | grep status`

Jeśli w wyniku zobaczysz "`status_module (shared)`", oznacza to, że moduł statusu jest już włączony. Jeśli nie, musisz włączyć moduł statusu.

Zastosuj szablon monitorowania Apache

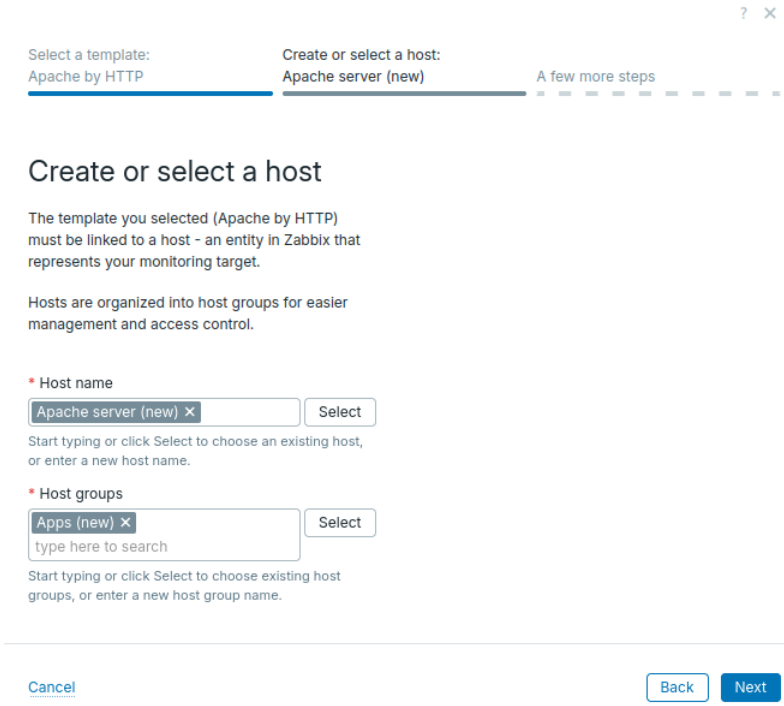
W Zabbix ten szablon jest gotowy do użycia. Zawiera wstępnie skonfigurowane elementy gotowe do zbierania danych, wizualizacji i analizy.

Aby zastosować szablon:

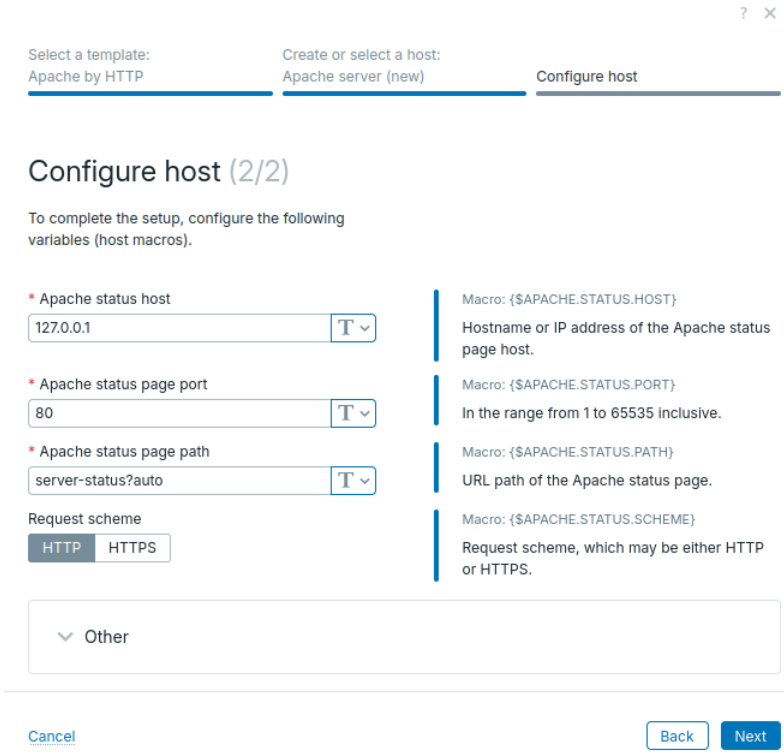
1. Uruchom Host Wizard w sekcji *Data collection* > *Hosts*.
2. Wybierz szablon *Apache by HTTP*.



3. Utwórz host Apache i dodaj do niego grupę. Ten host wirtualny będzie zawierał dane związane z Apache.



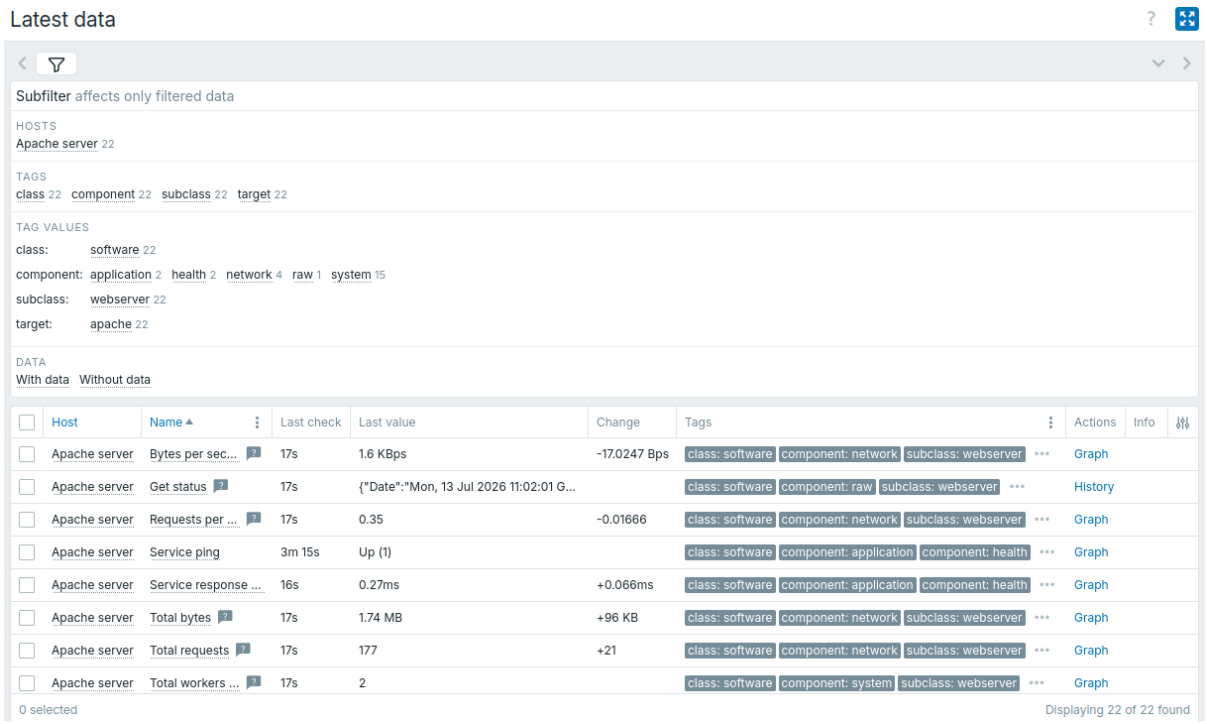
4. Skonfiguruj host. Co ważne, wprowadź adres serwera WWW Apache (127.0.0.1 na komputerze lokalnym, adres IP na komputerze zdalnym).



Pozostałe ustawienia możesz pozostawić domyślne.

5. Aby zakończyć pracę Host Wizard, kliknij *Create* i *Finish*.

Sprawdź najnowsze dane



Gratulacje, monitorowanie serwera WWW Apache zostało uruchomione!

Zabbix zapewnia opcje wizualizacji i dostarcza alerty, jeśli pojawią się jakiekolwiek problemy. Aby skonfigurować alerty w wiadomościach e-mail, zobacz Odbieranie powiadomień o problemach.

Monitorowanie serwera Linux

Aby rozpocząć monitorowanie serwera Linux, wykonaj następujące kroki:

1. Zastosuj szablon monitorowania Linux.
2. Zainstaluj agent Zabbix na serwerze.
3. Sprawdź najnowsze dane.

Aby rozpocząć monitorowanie zdalnego serwera Linux, musisz znać jego adres IP oraz adres IP serwera Zabbix.

Na serwerze Linux do monitorowania zostanie użyty agent Zabbix. Instrukcje instalacji agenta zostaną podane w dalszej części procesu.

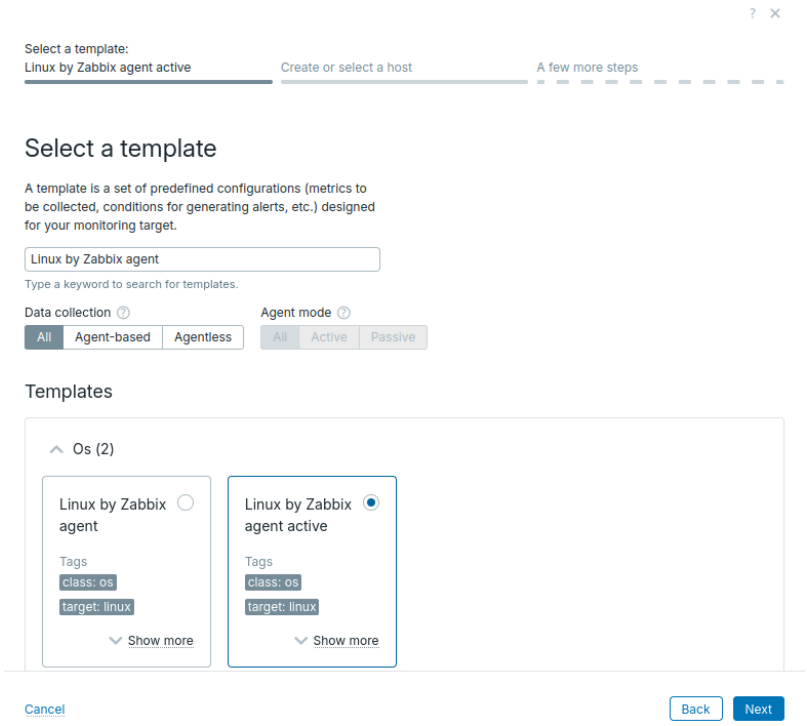
Aby skorzystać z tego przewodnika, musisz mieć zainstalowany serwer Zabbix oraz interfejs webowy installed. Musisz być zalogowany do Zabbix jako domyślny użytkownik *Admin*.

Zastosuj szablon monitorowania systemu Linux

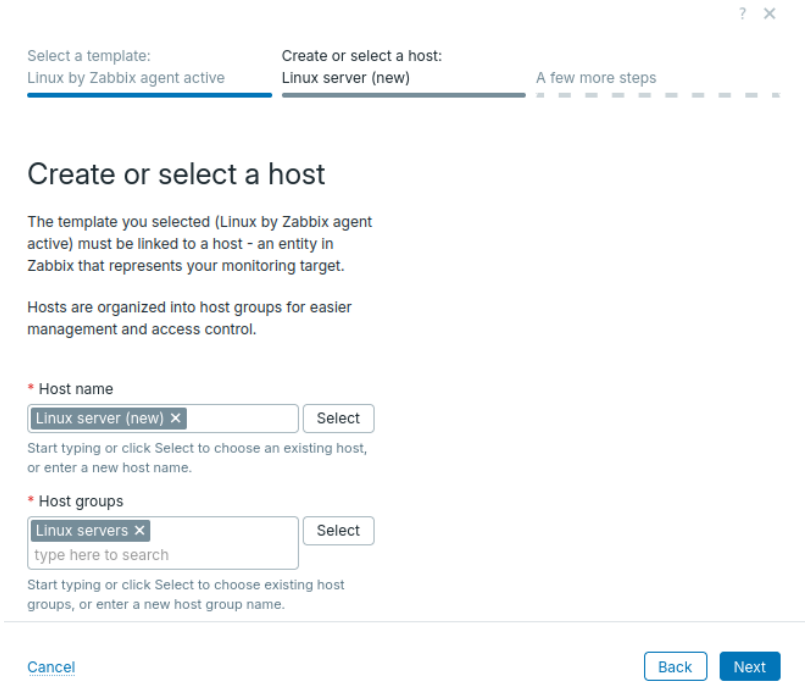
W Zabbix ten szablon jest gotowy do użycia. Zawiera wstępnie skonfigurowane elementy gotowe do zbierania danych, wizualizacji i analizy.

Aby zastosować szablon:

1. Uruchom Host Wizard w *Data collection* > *Hosts*.
2. Wybierz szablon *Linux by Zabbix agent active*.



3. Utwórz host dla serwera Linux i dodaj do niego grupę. Ten host wirtualny będzie zawierał wszystkie dane związane z monitorowaniem serwera Linux.



4. Skonfiguruj i zainstaluj agent.

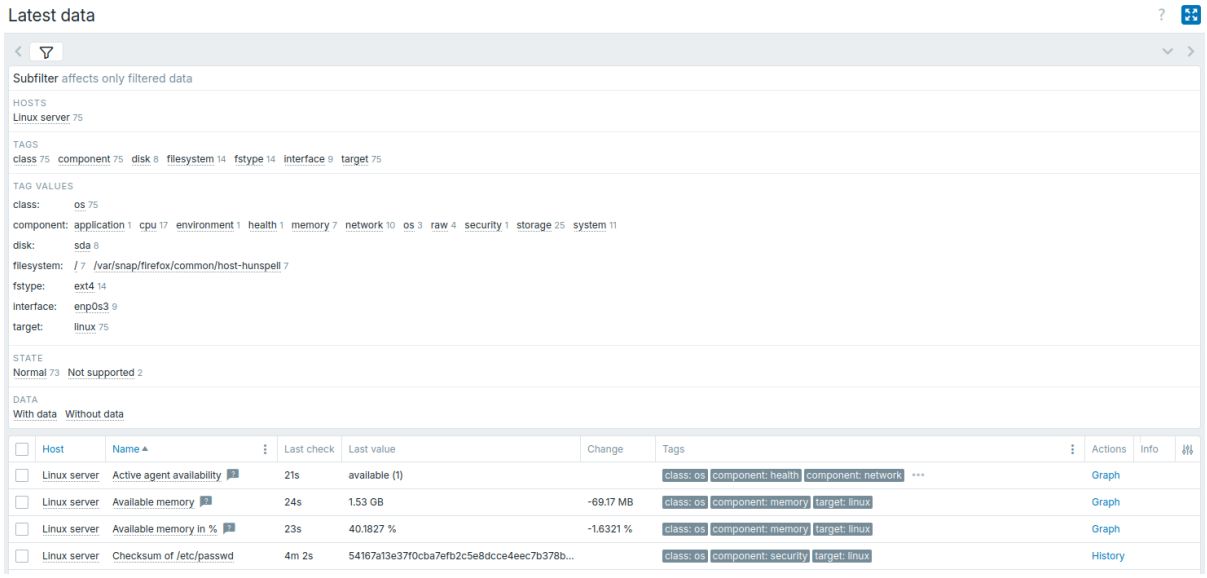


Aby zainstalować agent, wykonaj podane polecenie na serwerze Linux. Spowoduje ono instalację agenta oraz skonfiguruje połączenie agenta z serwerem Zabbix z użyciem szyfrowania.
 Jeśli agent jest już zainstalowany, możesz pominąć ten krok (kliknij *Next*). Zostaniesz jednak poproszony o podanie identyfikatora klucza współdzielonego oraz samego klucza współdzielonego, aby połączyć się z agentem.

Pozostałe ustawienia możesz pozostawić bez zmian.

5. Zakończ konfigurację hosta: kliknij *Create* i *Finish*.

Sprawdź najnowsze dane



Gratulacje, monitorowanie serwera Linux zostało uruchomione!

Zabbix oferuje opcje wizualizacji i dostarcza alerty, jeśli pojawią się jakiekolwiek problemy. Aby skonfigurować alerty w wiadomościach e-mail, zobacz Odbieranie powiadomień o problemach.

Monitorowanie serwera MySQL

Aby rozpocząć monitorowanie serwera MySQL, wykonaj następujące kroki:

1. Utwórz ograniczonego użytkownika MySQL.
2. Zastosuj szablon monitorowania MySQL.
3. Sprawdź najnowsze dane.

Aby rozpocząć monitorowanie serwera MySQL, musisz znać jego DSN oraz adres IP serwera Zabbix.

Do monitorowania serwera MySQL zostanie użyty agent Zabbix. Instrukcje instalacji agenta zostaną podane w trakcie procesu.

Aby skorzystać z tego przewodnika, musisz mieć zainstalowany serwer Zabbix oraz interfejs webowy installed. Musisz być zalogowany do Zabbix jako domyślny użytkownik *Admin*.

Utwórz użytkownika MySQL z ograniczonymi uprawnieniami

Zaleca się utworzenie użytkownika MySQL, którego uprawnienia będą ograniczone wyłącznie do tych wymaganych do monitorowania MySQL.
Zastąp <password> silnym, unikalnym hasłem.

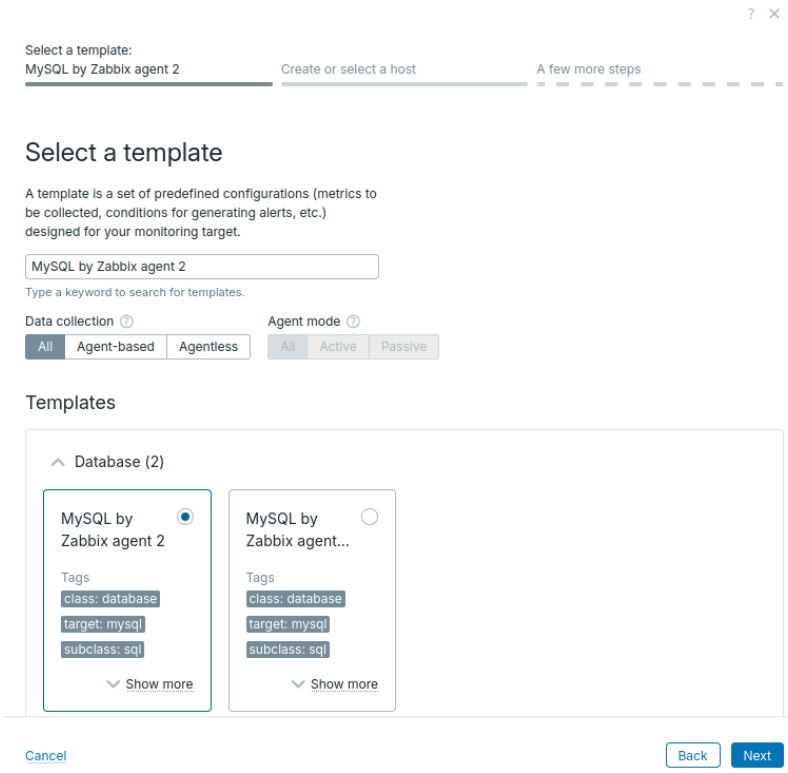
```
mysql> CREATE USER 'zbx_monitor'@'%' IDENTIFIED BY '<password>';
mysql> GRANT REPLICATION CLIENT,PROCESS,SHOW DATABASES,SHOW VIEW ON *.* TO 'zbx_monitor'@'%';
```

Zastosuj szablon monitorowania MySQL

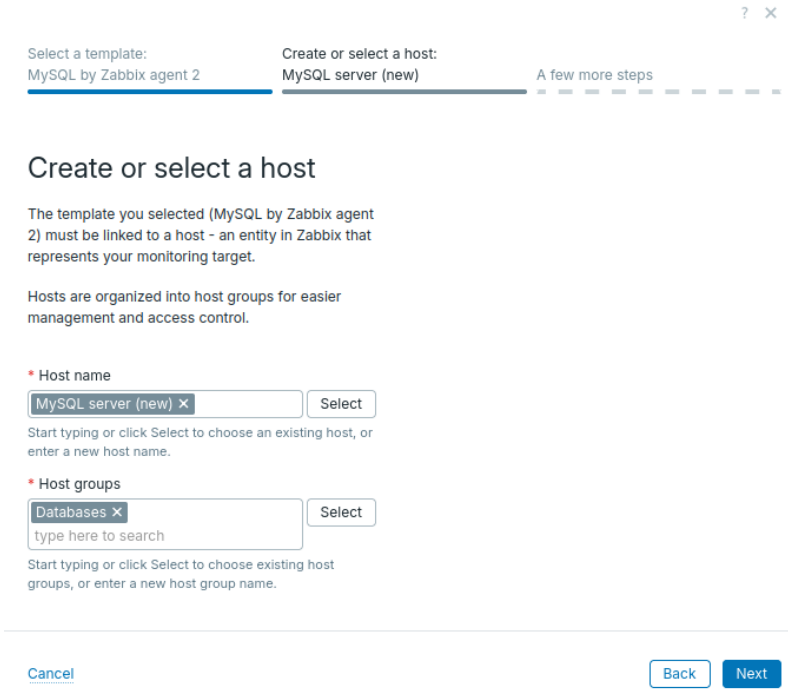
W Zabbix ten szablon jest gotowy do użycia. Zawiera wstępnie skonfigurowane elementy gotowe do zbierania danych, wizualizacji i analizy.

Aby zastosować szablon:

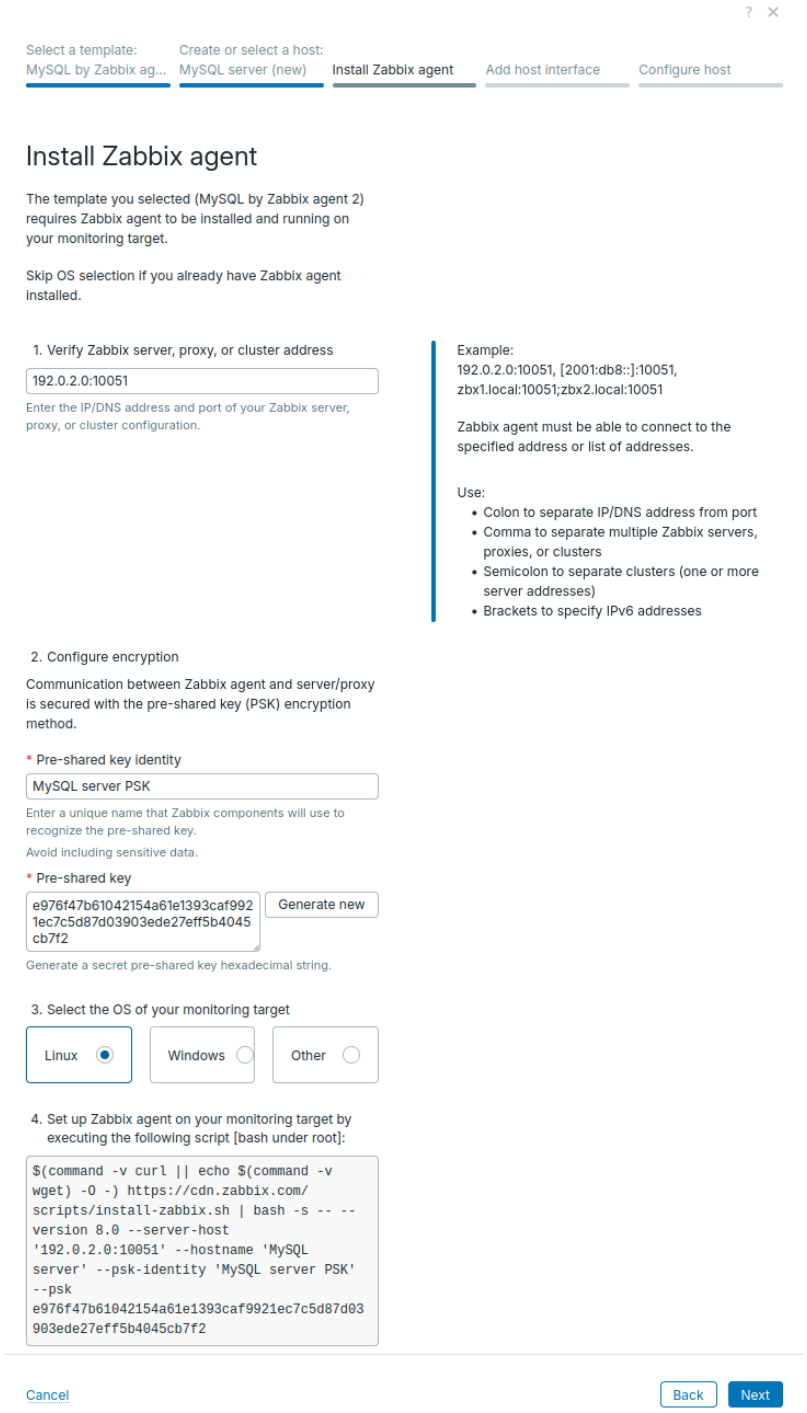
1. Uruchom Host Wizard w *Data collection* > *Hosts*.
2. Wybierz szablon *MySQL by Zabbix agent 2*.



3. Utwórz host serwera MySQL i dodaj do niego grupę. Ten host wirtualny będzie zawierał wszystkie dane związane z monitorowaniem serwera MySQL.

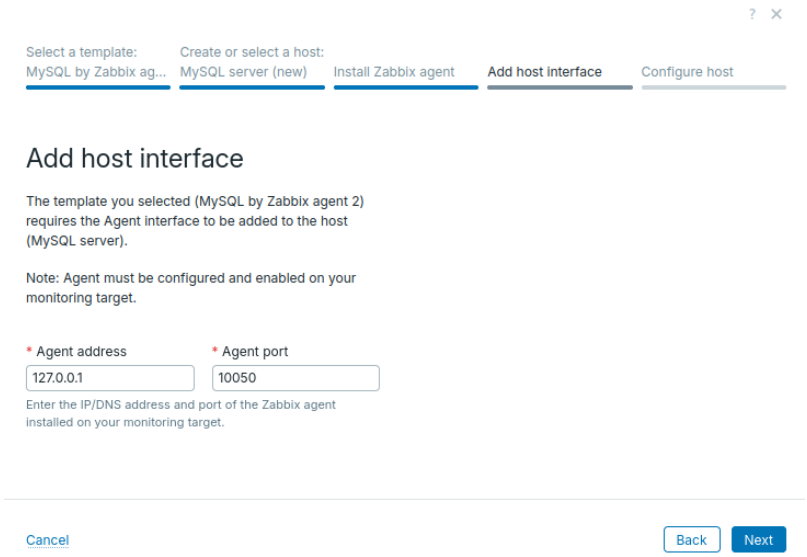


4. Skonfiguruj i zainstaluj agent 2 (Zabbix agent 2 zapewnia monitorowanie MySQL od razu po instalacji).

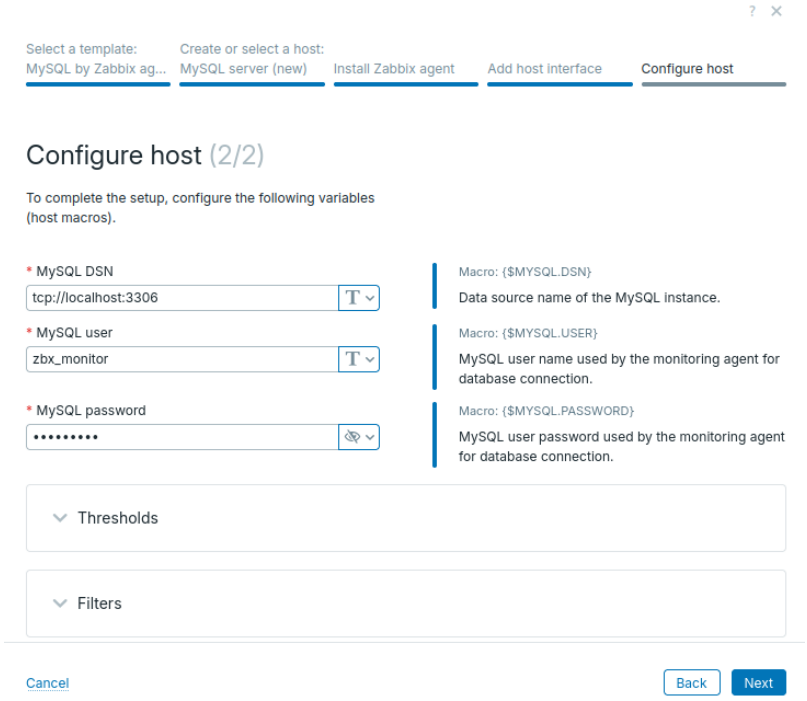


Aby zainstalować agenta, wykonaj podane polecenie na maszynie, na której działa serwer MySQL. Spowoduje to instalację agenta oraz skonfigurowanie połączenia agenta z serwerem Zabbix z użyciem szyfrowania.
 Jeśli agent 2 jest już zainstalowany, możesz pominąć ten krok (kliknij *Next*). Zostaniesz jednak poproszony o podanie tożsamości klucza wstępnie współdzielonego oraz klucza wstępnie współdzielonego do połączenia z agentem.

5. Wprowadź adres IP i port agenta (serwer Linux).



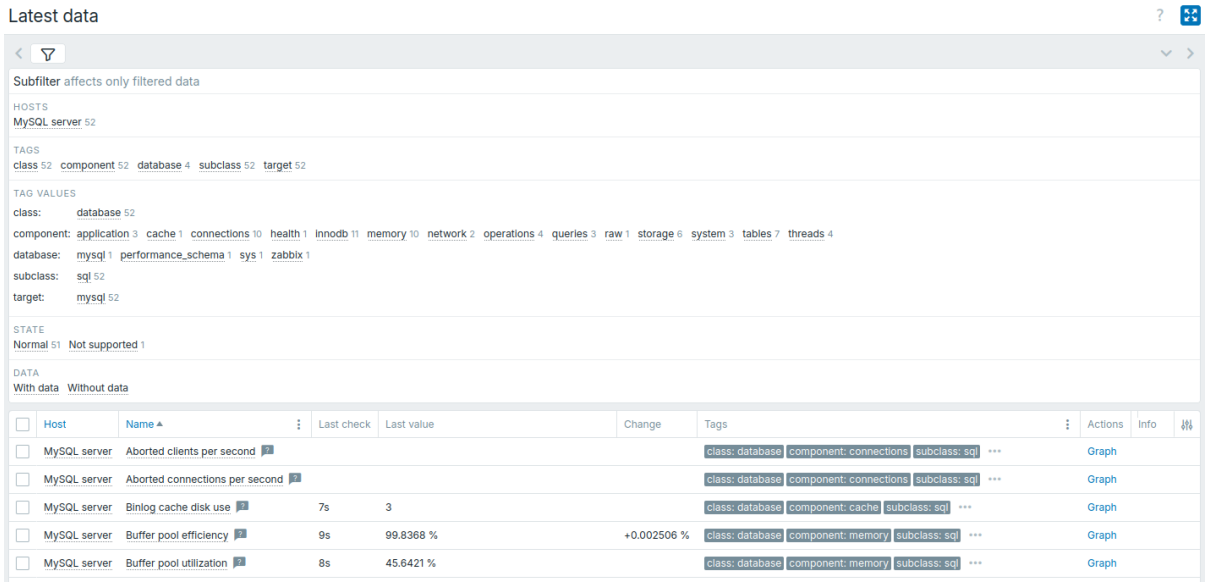
6. Zakończ konfigurację hosta. Co ważne, zostaniesz poproszony o podanie DSN serwera MySQL, użytkownika MySQL i hasła. Podaj użytkownika `zbx_monitor` wraz z utworzonym hasłem.



Pozostałe ustawienia możesz pozostawić jako domyślne.

6. Aby zakończyć Host Wizard, kliknij *Create* i *Finish*.

Sprawdź najnowsze dane



Gratulacje, monitorowanie serwera MySQL zostało uruchomione!

Zabbix zapewnia opcje wizualizacji i dostarcza alerty, jeśli pojawią się jakiekolwiek problemy. Aby skonfigurować alerty e-mail, zobacz Odbieranie powiadomień o problemach.

Więcej przewodników

Ta sekcja dokumentacji zawiera krótkie przepisy dotyczące konfiguracji Zabbixa do niektórych często wymaganych celów monitorowania.

Została zaprojektowana z myślą o nowych użytkownikach Zabbixa i może służyć jako przewodnik po innych sekcjach dokumentacji, które zawierają informacje potrzebne do rozwiązania danego zadania.

Dostępne są następujące przewodniki szybkiego odniesienia:

- Monitoruj Linux za pomocą agent Zabbix
- Monitoruj Windows za pomocą agent Zabbix
- Monitoruj Apache przez HTTP
- Monitoruj MySQL za pomocą agent Zabbix 2
- Monitoruj VMware za pomocą Zabbix
- Monitoruj ruch sieciowy za pomocą Zabbix
- Monitoruj ruch sieciowy za pomocą Zabbix, używając aktywnych kontroli
- Monitoruj witryny internetowe za pomocą pozycji przeglądarki
- Monitoruj certyfikaty witryn internetowych za pomocą agent Zabbix 2 (pasywnie)
- Monitoruj przełącznik sieciowy lub router za pomocą Zabbix
- Monitoruj dziennik zdarzeń Windows, używając aktywnych kontroli

Monitorowanie systemu Linux za pomocą agent Zabbix

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania maszyn z systemem Linux za pomocą Zabbix. Opisane w tym samouczku kroki można zastosować do dowolnego systemu operacyjnego opartego na Linuksie.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbixa i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania Twojej maszyny z systemem Linux. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Configuration w podręczniku Zabbixa.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

Instalacja Zabbix agent

Zabbix agent to proces odpowiedzialny za zbieranie danych.

Sprawdź wersję swojego serwera Zabbix:

```
zabbix_server -V
```

Zainstaluj Zabbix agent w tej samej wersji (zalecane) na komputerze z systemem Linux, który chcesz monitorować. W zależności od potrzeb monitorowania może to być ten sam komputer, na którym zainstalowano serwer Zabbix, albo zupełnie inny komputer.

Wybierz najbardziej odpowiednią metodę instalacji:

- Uruchom jako kontener Docker - zobacz listę dostępnych obrazów w repozytorium Zabbix Docker.
- Zainstaluj z pakietów Zabbix (dostępnych dla Alma Linux, CentOS, Debian, Oracle Linux, Raspberry Pi OS, RHEL, Rocky Linux, SUSE Linux Enterprise Server, Ubuntu).
- Skompiluj ze źródeł.

Skonfiguruj Zabbix do monitorowania

agent Zabbix może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie).

> **Note:**
> Sprawdzenie pasywne to proste żądanie danych. serwer Zabbix lub proxy pyta o określone dane (na przykład obciążenie CPU), a agent Zabbix odsyła wynik do serwera. Sprawdzenia aktywne wymagają bardziej złożonego przetwarzania. Agent musi najpierw pobrać z serwera(ów) listę pozycji do niezależnego przetwarzania, a następnie zbiorczo odesłać dane. Więcej informacji znajdziesz w sekcji Sprawdzenia pasywne i aktywne agenta.

Szablony monitorowania dostarczane przez Zabbix zwykle oferują dwie alternatywy - szablon dla agenta Zabbix oraz szablon dla agenta Zabbix (aktywny). W przypadku pierwszej opcji agent będzie zbierał metryki w trybie pasywnym. Takie szablony zapewnią identyczne wyniki monitorowania, ale z użyciem różnych protokołów komunikacji.

Dalsza konfiguracja Zabbix zależy od tego, czy wybierzesz szablon dla sprawdzeń agenta Zabbix aktywnych czy pasywnych.

Kontrole pasywne

Zabbix agent

1. Otwórz plik konfiguracyjny agenta na maszynie, na której zainstalowano agenta.

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

2. Dodaj adres IP lub nazwę DNS (oraz opcjonalnie port) swojego serwera Zabbix do parametru serwera. Na przykład:

```
Server=192.0.2.1:10051
```

Zabbix agent użyje tego adresu, aby akceptować połączenia przychodzące wyłącznie z określonych serwerów Zabbix lub proxy w celu pobierania danych.

3. Uruchom ponownie Zabbix agent.

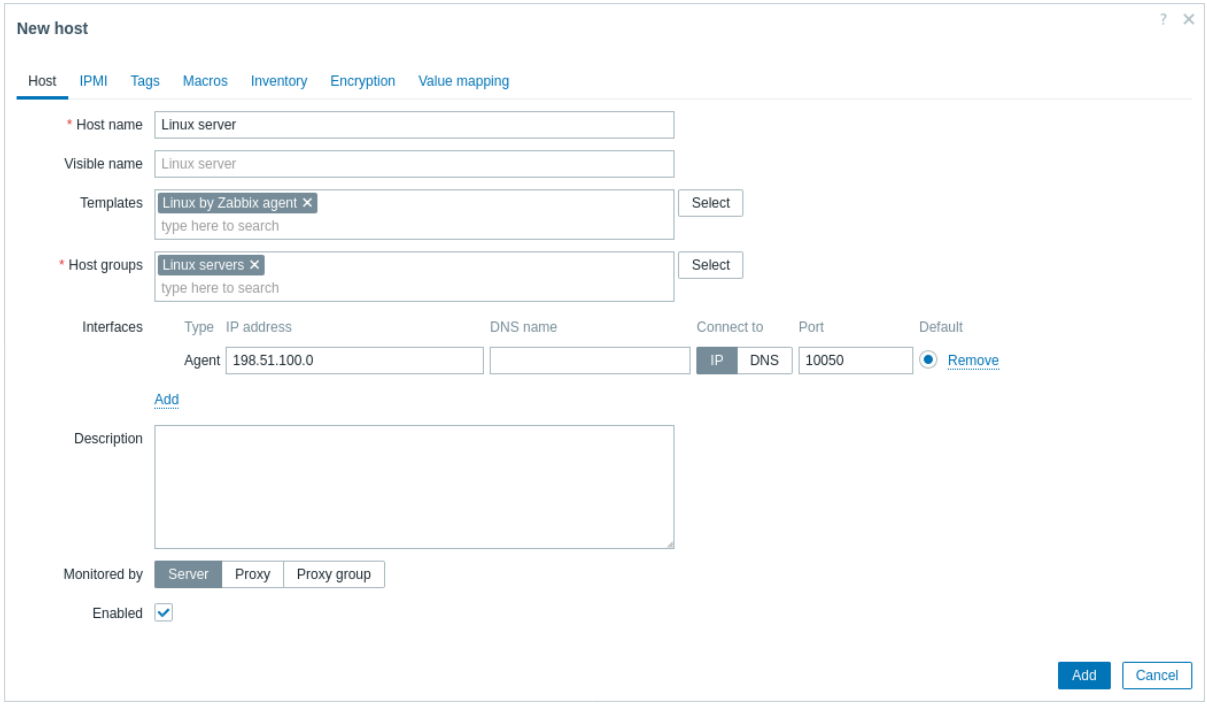
```
systemctl restart zabbix-agent
```

frontend Zabbix

1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie webowym Zabbix.

- W polu *Host name* wpisz nazwę hosta (np. ”Linux server”).
- W polu *Templates* wpisz lub wybierz szablon ”Linux by Zabbix agent”, który zostanie powiązany z hostem.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. ”Linux servers”).
- W parametrze *Interfaces* dodaj interfejs *Agent* i określ adres IP lub nazwę DNS komputera z systemem Linux, na którym zainstalowano agent.



3. Kliknij *Add*, aby dodać host. Ten host będzie reprezentował monitorowany komputer z systemem Linux.

Aktywne kontrole

Zabbix agent

1. Otwórz plik konfiguracyjny agenta na maszynie, na której jest zainstalowany agent:

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

2. Dodaj adres IP lub nazwę DNS (oraz opcjonalnie port) serwera Zabbix do parametru ServerActive. Na przykład:

```
ServerActive=192.0.2.1:10051
```

Zabbix agent użyje tego adresu, aby połączyć się z portem trappera serwera Zabbix (domyślnie: 10051) i zażądać danych konfiguracyjnych dotyczących aktywnych kontroli.

3. Zdefiniuj parametr Hostname, który musi odpowiadać nazwie hosta zdefiniowanej w frontend Zabbix. W naszym przykładzie jest to:

```
Hostname=Linux server
```

Wartość Hostname musi się zgadzać, ponieważ w przypadku aktywnych kontroli Zabbix agent używa jej do pobrania prawidłowej konfiguracji hosta z serwera. W szczególności agent inicjuje połączenie z serwerem i identyfikuje się za pomocą wartości Hostname. Następnie serwer udostępnia konfigurację monitorowania dla tego hosta. Jeśli wartości te się różnią, agent nie otrzyma odpowiedniej konfiguracji, co spowoduje brak metryk lub problemy z monitorowaniem.

4. Uruchom ponownie Zabbix agent.

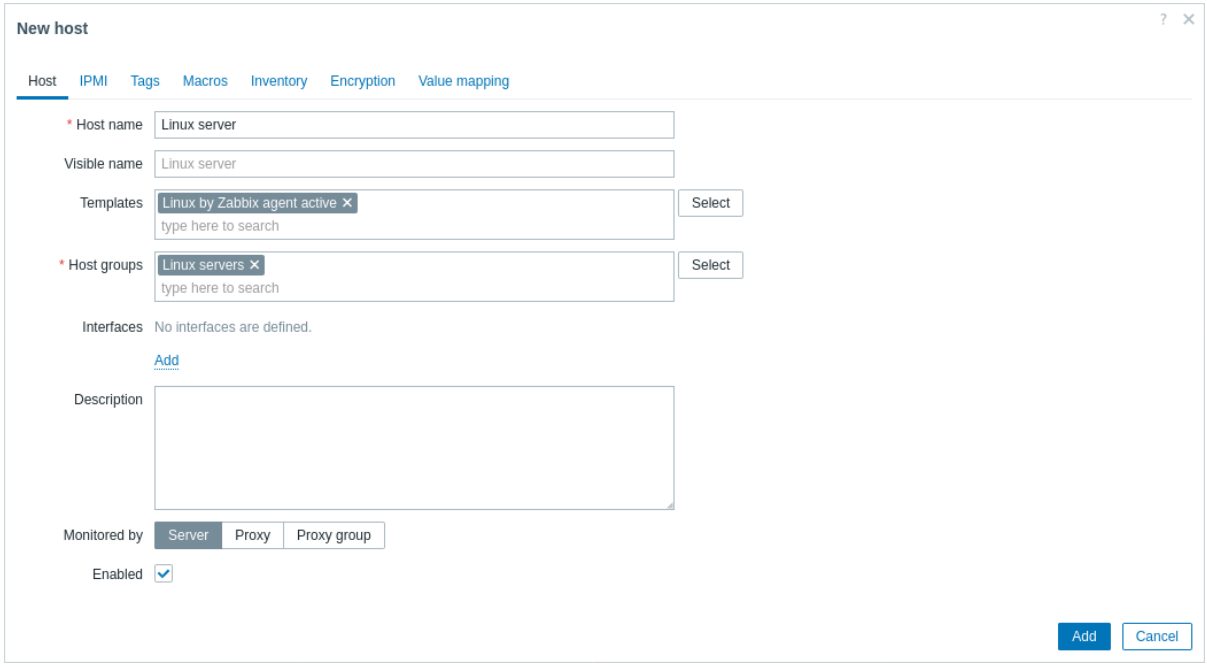
```
systemctl restart zabbix-agent
```

frontend Zabbix

1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie webowym Zabbix.

- W polu *Host name* wpisz nazwę hosta (np. ”Linux server”), która odpowiada wartości parametru *Hostname* zdefiniowanej wcześniej w pliku konfiguracyjnym agenta.
- W polu *Templates* wpisz lub wybierz szablon ”Linux by Zabbix agent active”, który zostanie powiązany z hostem.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. ”Linux servers”).
- Pozostaw *Interfaces* niezdefiniowane; interfejs nie jest potrzebny dla aktywnych kontroli, ponieważ agent inicjuje połączenie z serwerem zamiast nasłuchiwać na połączenie z serwera.

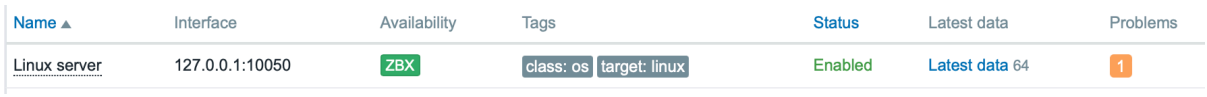


3. Kliknij *Add*, aby dodać host. Ten host będzie reprezentował monitorowaną maszynę z systemem Linux.

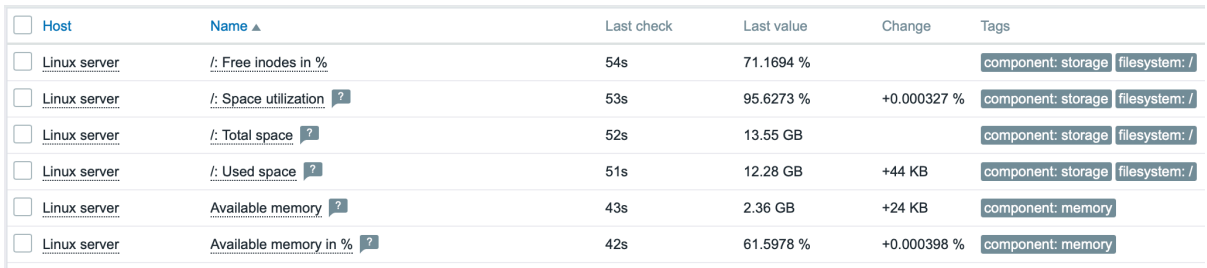
Wyświetlanie zebranych metryk

Gratulacje! Na tym etapie Zabbix już monitoruje Twój komputer z systemem Linux.

Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring* > *Hosts* menu section i kliknij *Latest data* obok hosta.



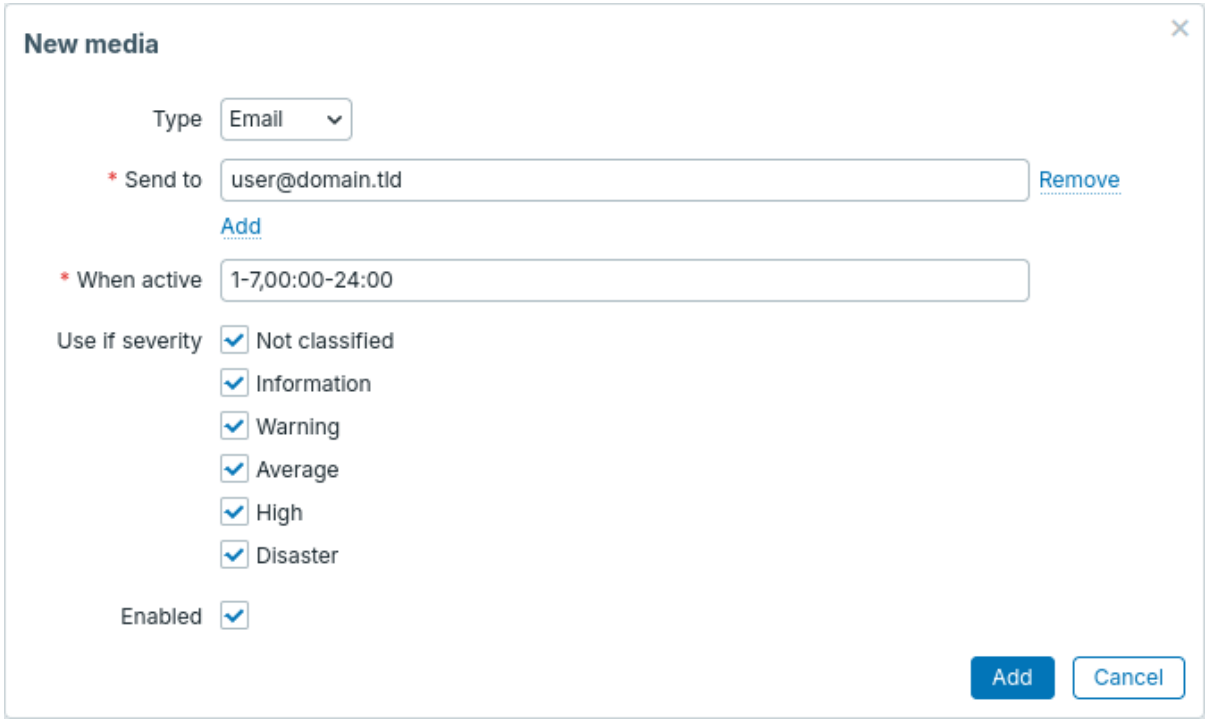
Ta akcja otworzy listę wszystkich najnowszych metryk zebranych z hosta serwera Linux.



Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemie z Twoją infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika* > *Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem dotyczącym otrzymywania powiadomień o problemach.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

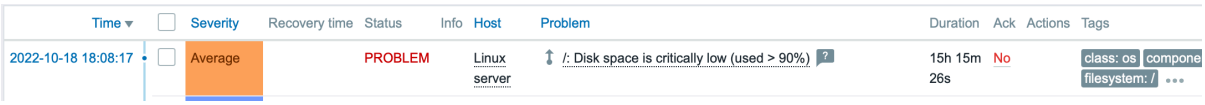
Przetestuj swoją konfigurację

W systemie Linux możesz zasymulować duże obciążenie CPU, a w rezultacie otrzymać alert o problemie, uruchamiając:

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

Może być konieczne uruchomienie kilku procesów md5sum, aby obciążenie CPU przekroczyło próg.

Gdy Zabbix wykryje problem, pojawi się on w sekcji *Monitoring* > *Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

Zobacz także:

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk (monitorowanie niestandardowe bez szablonów).
- Pozycje agenta Zabbix, Pozycje agenta Zabbix dla Windows - pełna lista metryk, które można monitorować za pomocą agenta Zabbix w systemie Windows.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Instalacja z pakietów - jak instalować komponenty Zabbix przy użyciu oficjalnych pakietów RPM i DEB dla różnych dystrybucji systemu Linux, zapewniając dostęp do najnowszych funkcji i poprawek błędów.

Monitorowanie systemu Windows za pomocą agent Zabbix

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania maszyn z systemem Windows za pomocą Zabbix.

Dla kogo jest przeznaczony ten przewodnik

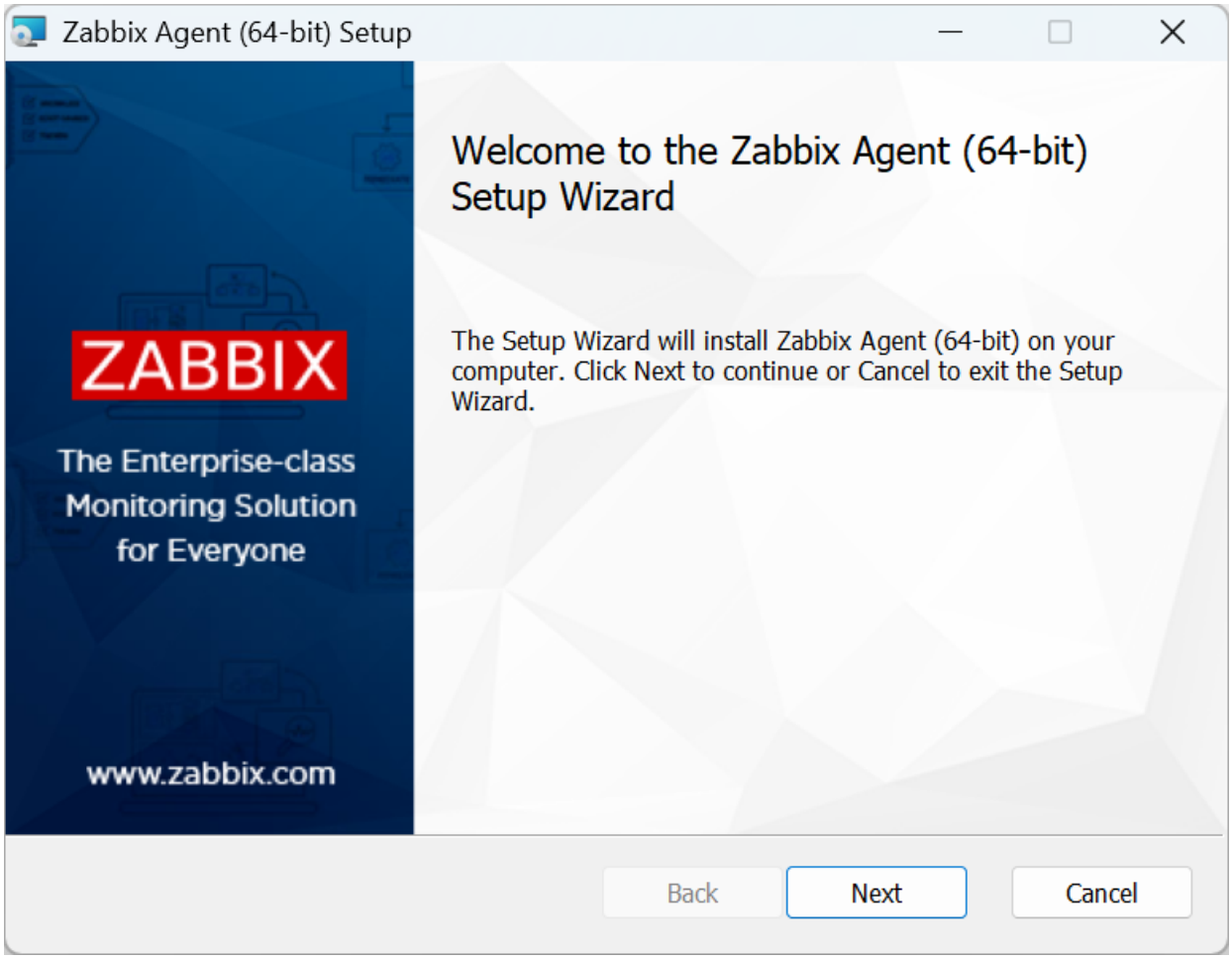
Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania maszyny z systemem Windows. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej zaawansowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed rozpoczęciem korzystania z tego przewodnika instalacji musisz pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dotyczącymi Twojego systemu operacyjnego.

Instalacja Zabbix agent

Zabbix agent to proces odpowiedzialny za zbieranie danych. Należy go zainstalować na komputerze z systemem Windows, który chcesz monitorować. Postępuj zgodnie z instrukcjami instalacji Zabbix agent dla Windows.



Skonfiguruj Zabbix do monitorowania

Agent Zabbix może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie).

Note:
Pasywne sprawdzenie to proste żądanie danych. Serwer Zabbix lub proxy pyta o określone dane (na przykład obciążenie CPU), a agent Zabbix odsyła wynik do serwera. Aktywne sprawdzenia wymagają bardziej złożonego przetwarzania. Agent musi najpierw pobrać z serwera(ów) listę pozycji do niezależnego przetwarzania, a następnie zbiorczo odesłać dane. Więcej informacji można znaleźć w sekcji Pasywne i aktywne sprawdzenia agenta.

Szablony monitorowania dostarczane przez Zabbix zwykle oferują dwie alternatywy - szablon dla agenta Zabbix oraz szablon dla agenta Zabbix (aktywny). W przypadku pierwszej opcji agent będzie zbierał metryki w trybie pasywnym. Takie szablony zapewnią identyczne wyniki monitorowania, ale z użyciem różnych protokołów komunikacji.

Dalsza konfiguracja Zabbix zależy od tego, czy wybierzesz szablon dla aktywnych czy pasywnych sprawdzeń agenta Zabbix.

Kontrole pasywne

frontend Zabbix

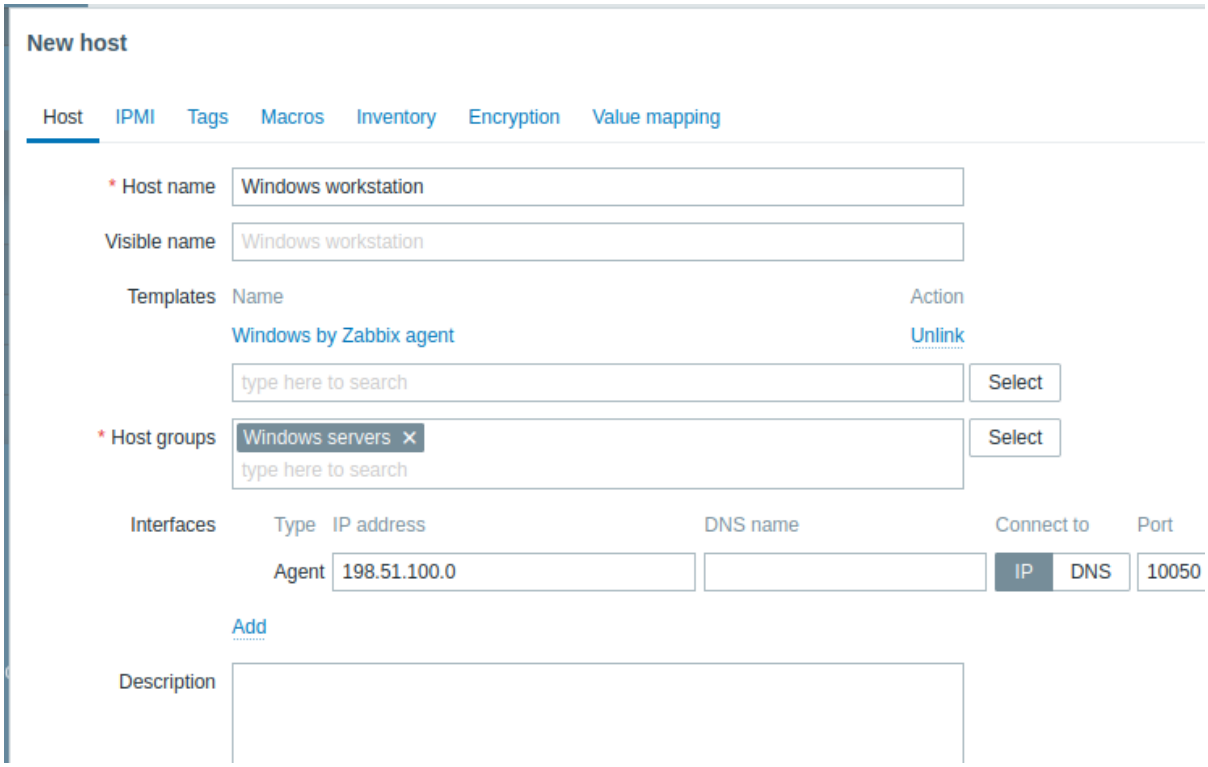
1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie web frontend Zabbix.

Ten host będzie reprezentował Twoją maszynę z systemem Windows.

3. W parametrze *Interfaces* dodaj interfejs *Agent* i określ adres IP lub nazwę DNS maszyny z systemem Windows, na której zainstalowano agent.

4. W parametrze *Templates* wpisz lub wybierz *Windows by Zabbix agent*.



Zabbix agent

W przypadku kontroli pasywnych Zabbix agent musi znać adres IP lub nazwę DNS serwera Zabbix. Jeśli podczas instalacji agenta podano prawidłowe informacje, plik konfiguracyjny został już zaktualizowany. W przeciwnym razie należy określić je ręcznie. Otwórz plik konfiguracyjny agenta i dodaj adres IP/DNS serwera Zabbix do parametru `Server`.

Przykład:

```
Server=192.0.2.22
```

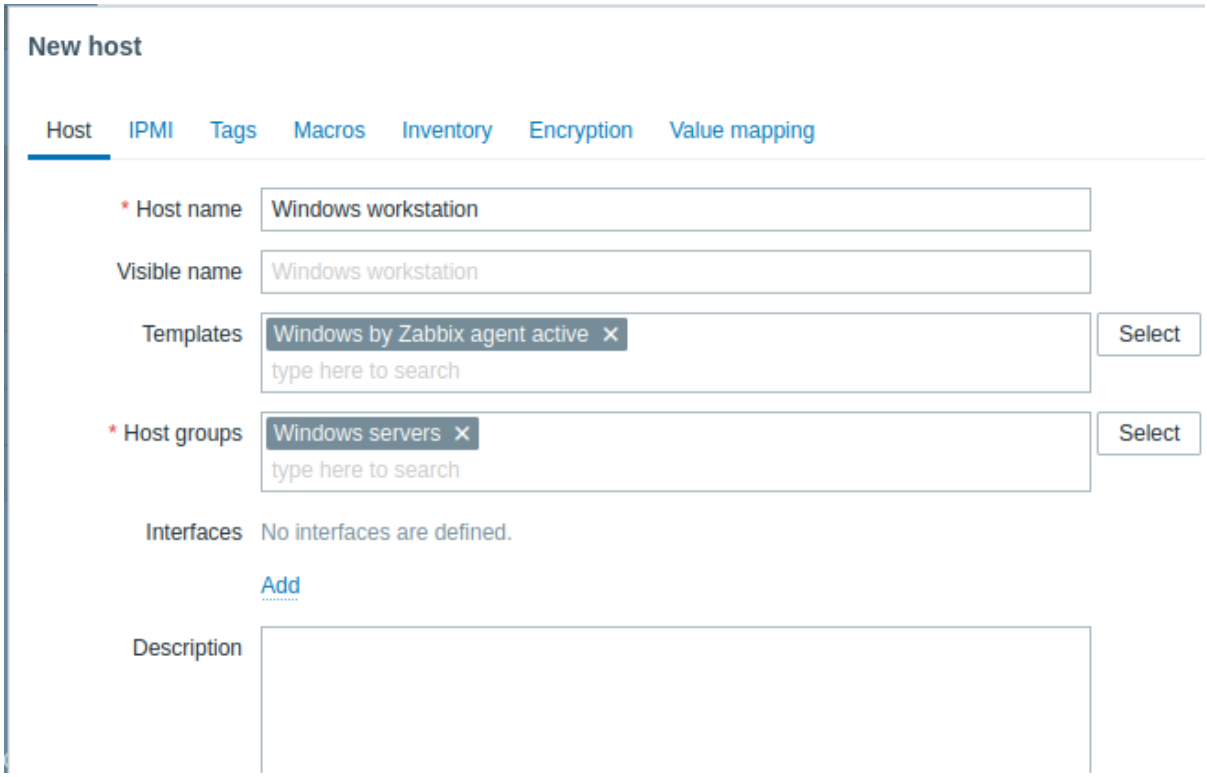
Aktywne sprawdzenia

frontend Zabbix

1. Zaloguj się do frontend Zabbix.
2. Utwórz host w interfejsie web Zabbix.

Ten host będzie reprezentować Twój komputer z systemem Windows.

3. W parametrze *Templates* wpisz lub wybierz *Windows by Zabbix agent active*.



Agent Zabbix

W folderze `C:\Program files\Zabbix Agent` otwórz plik `zabbix_agentd.conf` i dodaj:

- Nazwę hosta utworzonego w frontendzie Zabbix do parametru *Hostname*.
- Adres IP lub nazwę DNS komputera z serwerem Zabbix do parametru *ServerActive* (może być już wypełniony, jeśli został podany podczas konfiguracji agenta Zabbix).

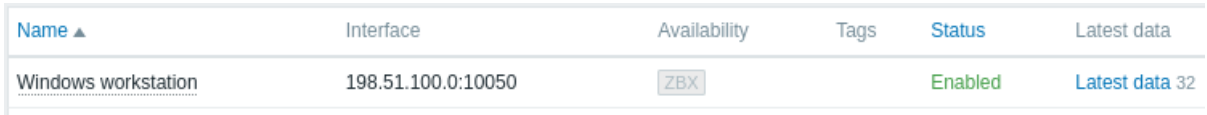
Przykład:

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

Wyświetlanie zebranych metryk

Gratulacje! W tym momencie Zabbix już monitoruje Twój komputer z systemem Windows.

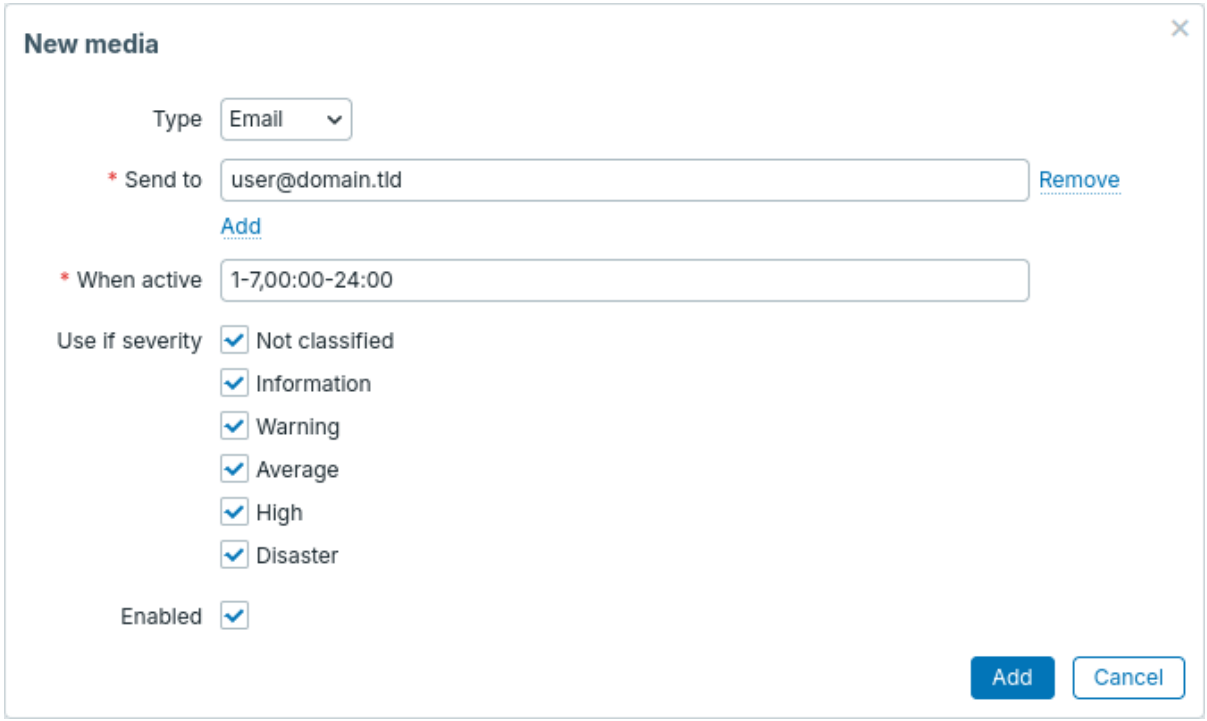
Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring* > *Hosts* menu section i kliknij *Latest data* obok hosta.



Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika* > *Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem dotyczącym otrzymywania powiadomień o problemach.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Note:
W systemie Windows możesz użyć narzędzia CpuStres, aby zasymulować wysokie obciążenie CPU i w rezultacie otrzymać alert o problemie.

Zobacz także:

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk (monitorowanie niestandardowe bez szablonów).
- Pozycje agenta Zabbix, Pozycje agenta Zabbix dla Windows - pełna lista metryk, które można monitorować za pomocą agenta Zabbix w systemie Windows.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).

Monitorowanie Apache przez HTTP

Wprowadzenie

Ta strona pokazuje szybki i prosty sposób rozpoczęcia monitorowania serwera WWW Apache bez instalowania jakiegokolwiek dodatkowego oprogramowania.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania instalacji Apache. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika instalacji musisz pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla swojego systemu operacyjnego.

Przygotowanie Apache

1. Sprawdź, której wersji Apache używasz:

W systemie opartym na RHEL uruchom:

```
httpd -v
```

W Debian/Ubuntu uruchom:

```
apache2 -v
```

2. Upewnij się, że moduł Status module jest włączony w Twojej instancji Apache.

W systemie opartym na RHEL uruchom:

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

W Debian/Ubuntu uruchom:

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

Jeśli nie widzisz status_module na liście, włącz moduł, uruchamiając:

W systemie opartym na RHEL uruchom:

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

W Debian/Ubuntu uruchom:

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

3. Edytuj plik konfiguracyjny Apache, aby zezwolić na dostęp do raportów statusu z adresu IP serwera Zabbix.

W systemie opartym na RHEL: `/etc/httpd/conf.modules.d/status.conf`:

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

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

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

Dodaj do pliku następujące wiersze (**zastąp 198.51.100.255** adresem IP swojego serwera Zabbix):

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

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

4. Uruchom ponownie Apache

W systemie opartym na RHEL uruchom:

```
sudo systemctl restart httpd
```

W Debian/Ubuntu uruchom:

```
sudo systemctl restart apache2
```

5. Aby sprawdzić, czy wszystko jest poprawnie skonfigurowane, uruchom (**zastąp 198.51.100.255** adresem IP swojego serwera Zabbix):

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

Odpowiedź powinna zawierać statystyki serwera WWW Apache.

Skonfiguruj Zabbix do monitorowania

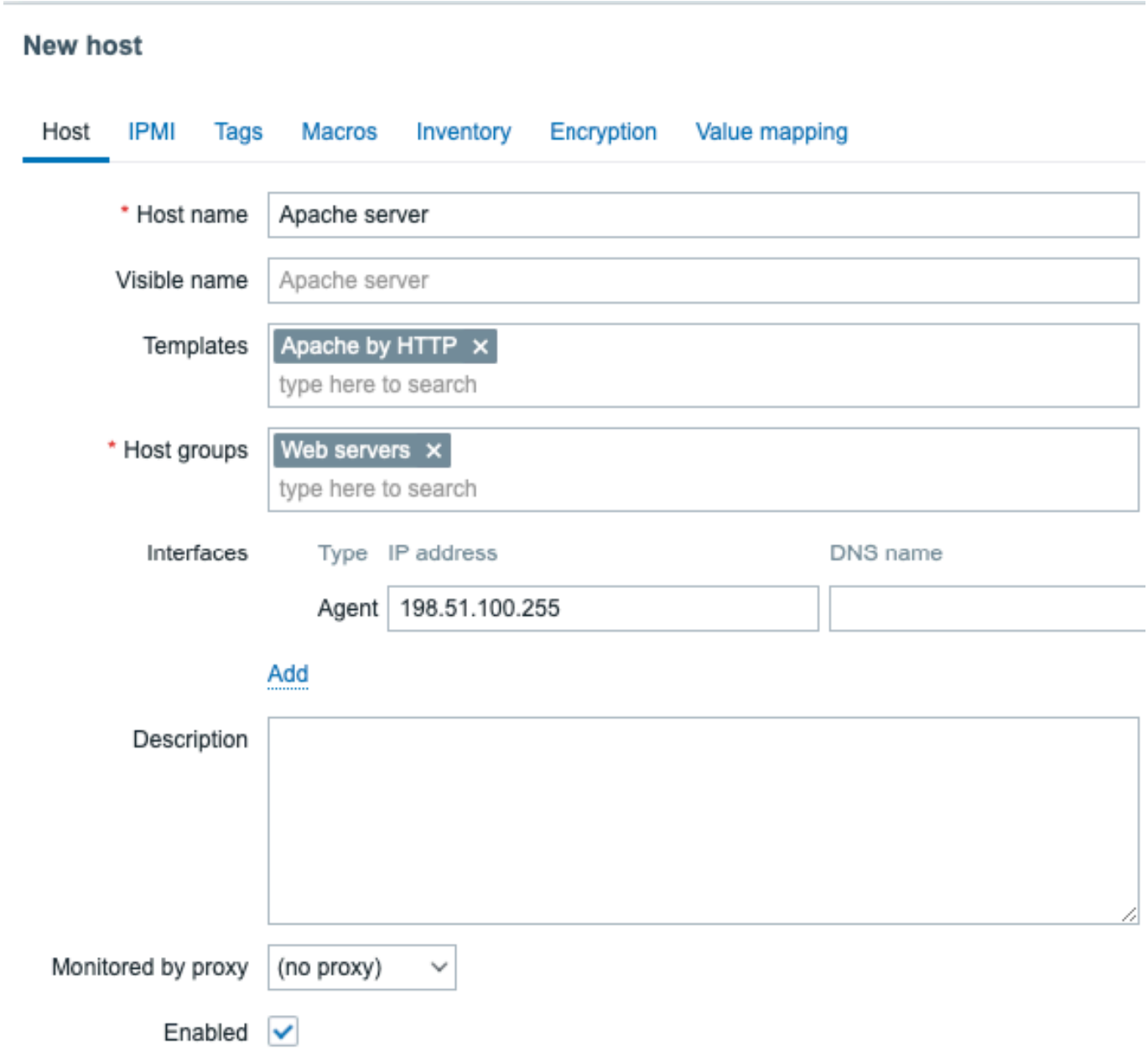
1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie webowym Zabbix.

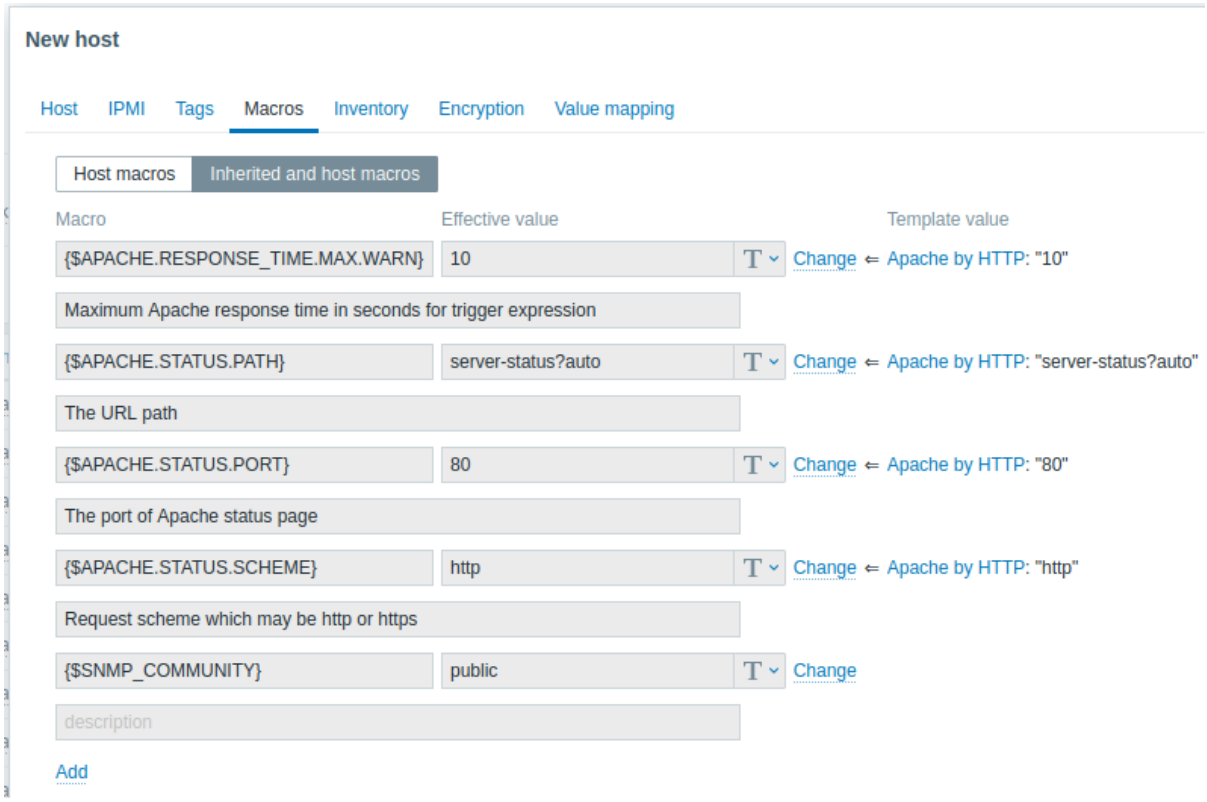
Ten host będzie reprezentował Twój serwer Apache.

3. W parametrze *Interfaces* dodaj interfejs *Agent* i określ adres IP swojej instancji Apache.
Nie musisz instalować agenta Zabbix na tej maszynie, interfejs będzie używany wyłącznie do rozwiązywania makra {HOST.CONN}.
To makro jest używane w pozycjach szablonu do lokalizowania instancji Apache.

4. W parametrze *Templates* wpisz lub wybierz *Apache by HTTP*.



5. Przejdź do zakładki **Macros** i wybierz tryb *Inherited and host macros*. Sprawdź, czy wartości makr {$APACHE.STATUS.PORT} i {$APACHE.STATUS.SCHEME} odpowiadają ustawieniom Twojej instalacji. Domyślnie port to 80, a schemat to http. Zmień wartości makr, jeśli używasz innego portu i/lub schematu.



Wyświetlanie zebranych metryk

Gratulacje! Na tym etapie Zabbix już monitoruje Twój serwer WWW Apache.

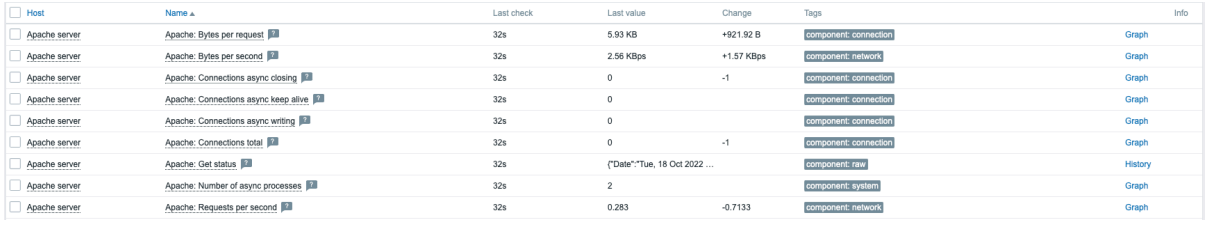
Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring* > *Hosts* menu section i kliknij *Dashboards* obok hosta.



Ta akcja przeniesie Cię do dashboardu hosta z najważniejszymi metrykami zebranymi ze strony /server-status Apache.



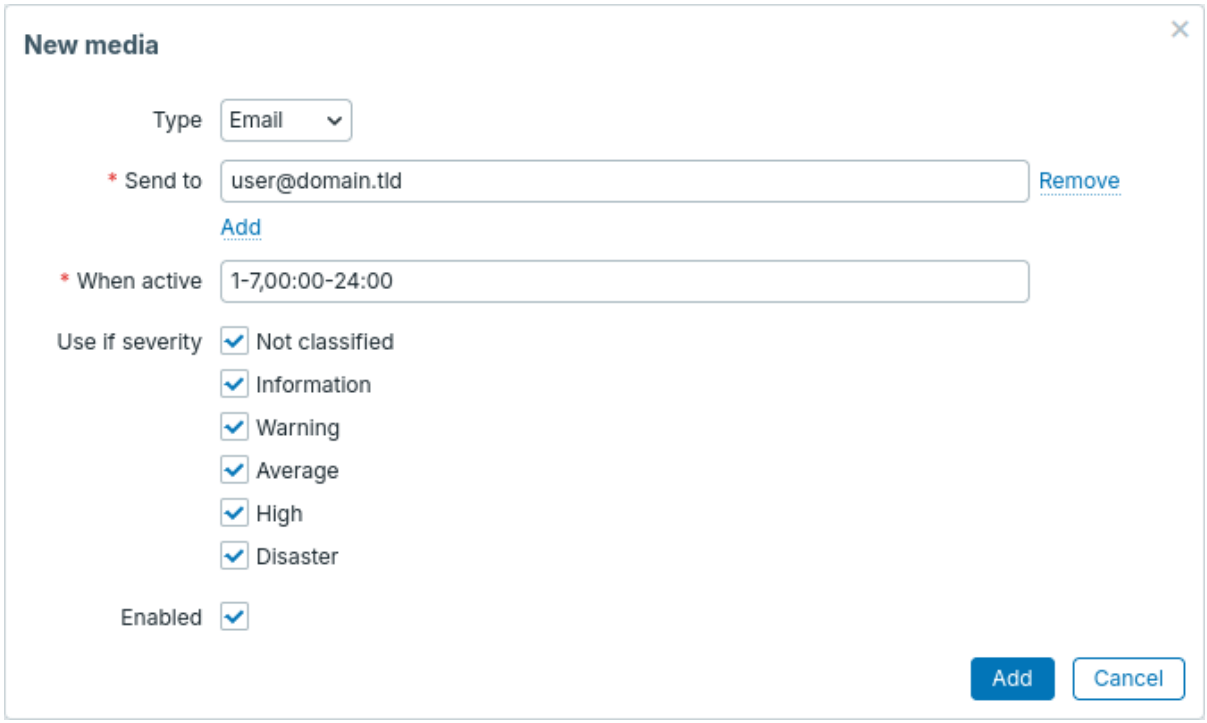
Alternatywnie, w *Monitoring* > *Hosts* możesz kliknąć *Latest data*, aby wyświetlić wszystkie najnowsze zebrane metryki na liście.



Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika* > *Profil*, przejdź do karty *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem dotyczącym otrzymywania powiadomień o problemach.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby zasymulować rzeczywisty problem i otrzymać testowe powiadomienie o problemie:

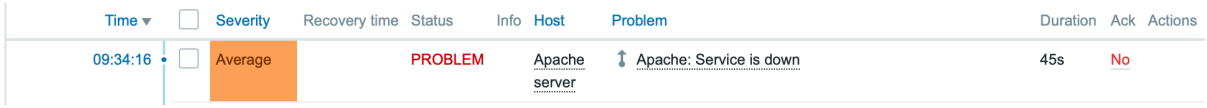
1. Otwórz konfigurację hosta *Apache server* w Zabbix.

2. Przejdź do zakładki Macros i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok makra {$APACHE.STATUS.PORT} i ustaw inny port.

4. Kliknij *Update*, aby zapisać konfigurację hosta.

5. Po kilku minutach Zabbix wykryje problem *Apache service is down*, ponieważ nie będzie już mógł połączyć się z instancją. Pojawi się on w sekcji *Monitoring* > *Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Zmień wartość makra z powrotem, aby rozwiązać problem i kontynuować monitorowanie Apache.

Zobacz także:

- Wzmocnienie zabezpieczeń serwera WWW - zalecane ustawienia zwiększające bezpieczeństwo serwera WWW.
- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Pozycje HTTP - jak monitorować niestandardowe metryki za pomocą agenta HTTP.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).

Monitorowanie MySQL za pomocą agent 2 Zabbix

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania serwera MySQL.

Aby monitorować serwer MySQL, dostępnych jest kilka podejść: agent Zabbix, Zabbix agent 2 lub standard Open Database Connectivity (ODBC). Główny nacisk tego przewodnika położono na monitorowanie serwera MySQL za pomocą Zabbix agent 2, co jest **zalecanym** podejściem ze względu na bezproblemową konfigurację w różnych środowiskach. Jednak na tej stronie znajdują się również instrukcje dotyczące innych podejść, więc możesz wybrać to, które najlepiej odpowiada Twoim wymaganiom.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania serwera MySQL. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent 2 Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

W zależności od środowiska niektóre kroki opisane w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku działającym pod kontrolą Ubuntu.

Utwórz użytkownika MySQL

Aby monitorować serwer MySQL, Zabbix wymaga dostępu do niego i jego procesów. Twoja instalacja MySQL ma już użytkownika z wymaganym poziomem dostępu (użytkownika ”zabbix”, który został utworzony podczas instalacji Zabbix), jednak ten użytkownik ma więcej uprawnień, niż jest to potrzebne do prostego monitorowania (uprawnienia do DROP baz danych, DELETE wpisów z tabel itp.). Dlatego należy utworzyć użytkownika MySQL wyłącznie do monitorowania serwera MySQL.

1. Połącz się z klientem MySQL, utwórz użytkownika ”zbx_monitor” (zastąp *<password>* dla użytkownika ”zbx_monitor” hasłem według własnego wyboru), i GRANT niezbędne uprawnienia temu użytkownikowi:

```
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;
```

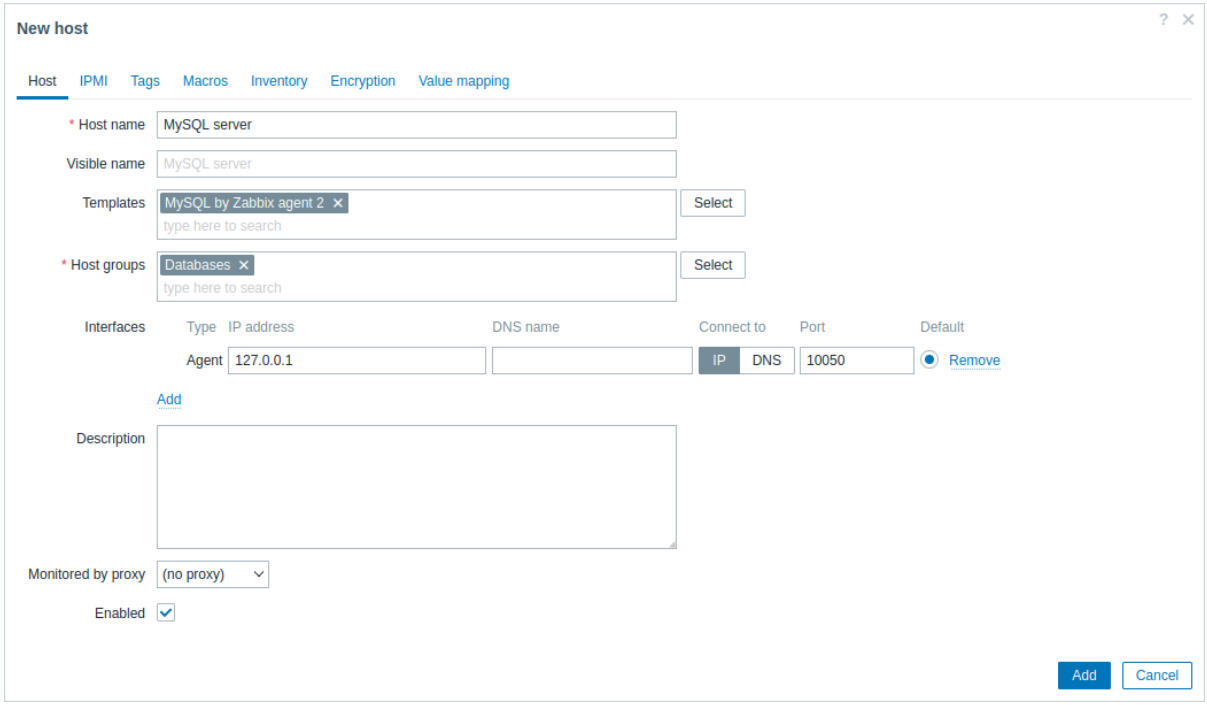
Po utworzeniu użytkownika możesz przejść do następnego kroku.

Skonfiguruj frontend Zabbix

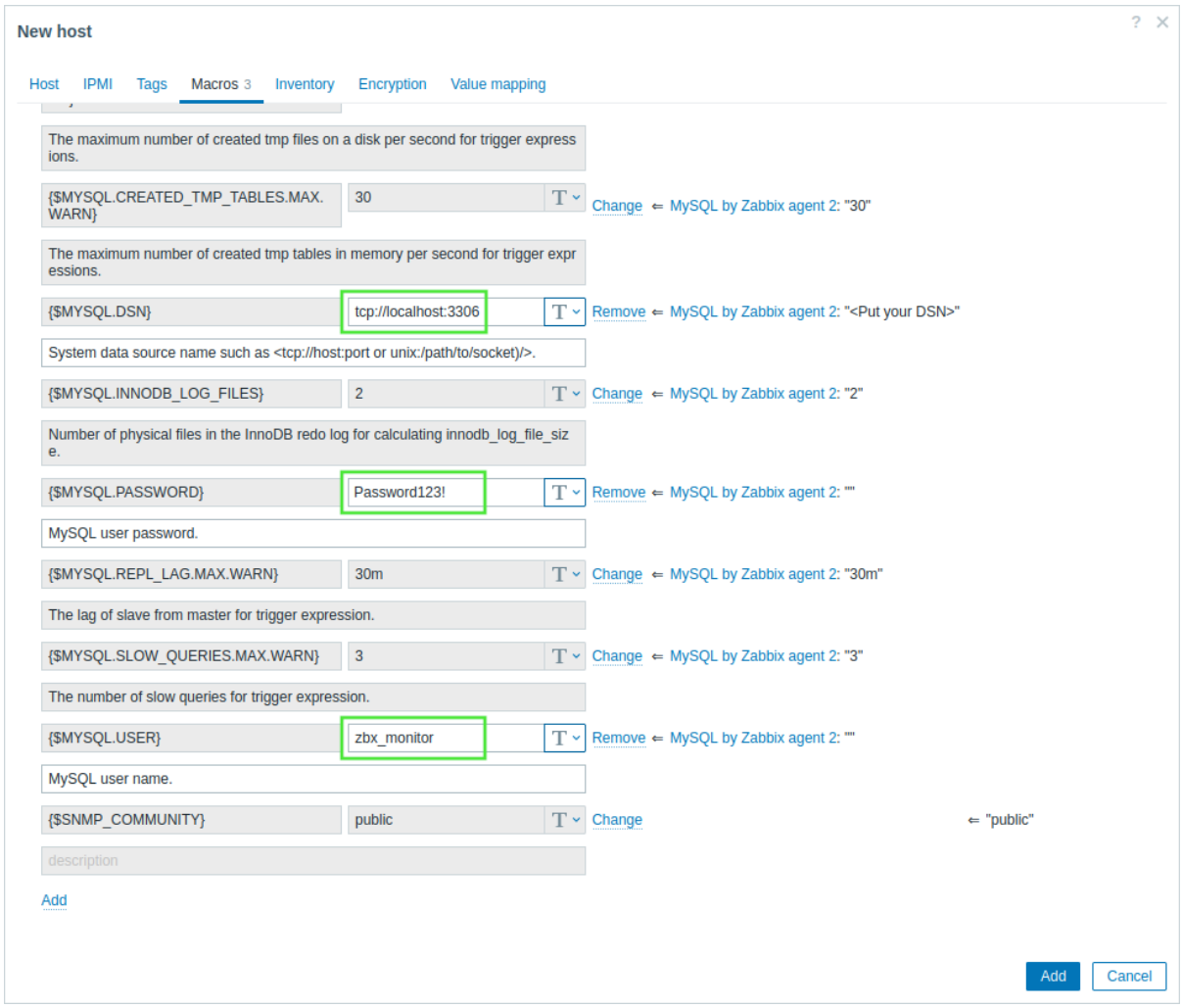
1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie webowym Zabbix:

   - W polu *Host name* wpisz nazwę hosta (np. ”MySQL server”).
   - W polu *Templates* wpisz lub wybierz szablon ”MySQL by Zabbix agent 2”, który zostanie powiązany z hostem.
   - W polu *Host groups* wpisz lub wybierz grupę hostów (np. ”Databases”).
   - W polu *Interfaces* dodaj interfejs typu ”Agent” i określ adres IP serwera MySQL. W tym przewodniku użyto ”127.0.0.1” (localhost) do monitorowania serwera MySQL zainstalowanego na tym samym komputerze co serwer Zabbix i agent Zabbix agent 2.



- Na karcie *Macros* przejdź do *Inherited and host macros*, znajdź następujące makra i kliknij *Change* obok wartości makra, aby ją zaktualizować:
  - {$MYSQL.DSN} - ustaw źródło danych serwera MySQL (ciąg połączenia nazwanej sesji](/manual/concepts/agent2/agent2_params_ z pliku konfiguracyjnego wtyczki MySQL Zabbix agent 2). W tym przewodniku użyto domyślnego źródła danych "tcp://localhost:3306" do monitorowania serwera MySQL zainstalowanego na tym samym komputerze co serwer Zabbix i agent Zabbix agent 2.
  - {$MYSQL.PASSWORD} - ustaw hasło wcześniej utworzonego użytkownika MySQL "zbx_monitor".
  - {$MYSQL.USER} - ustaw nazwę wcześniej utworzonego użytkownika MySQL "zbx_monitor".

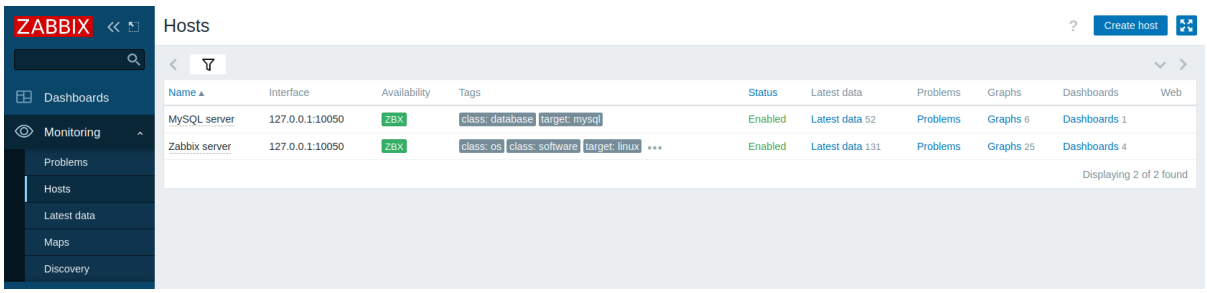


3. Kliknij *Add*, aby dodać host. Ten host będzie reprezentował Twój serwer MySQL.

Wyświetlanie zebranych metryk

Gratulacje! W tym momencie Zabbix już monitoruje Twój serwer MySQL.

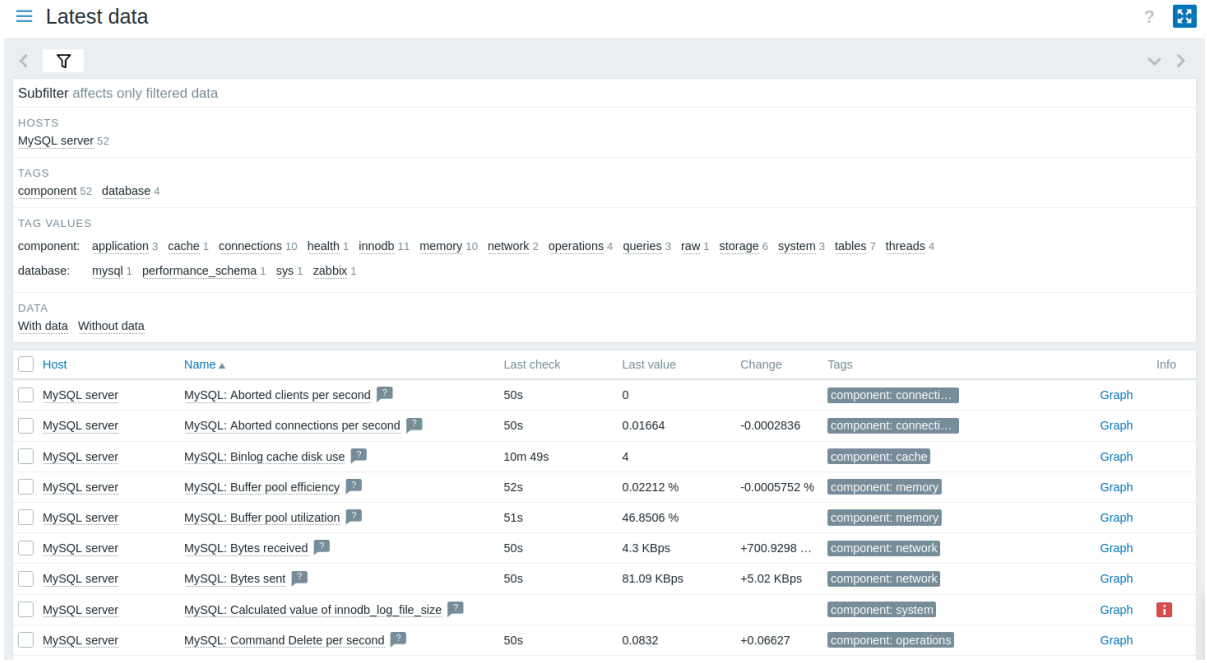
Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring > Hosts* i kliknij *Dashboards* obok hosta.



Ta akcja przeniesie Cię do dashboardu hosta (skonfigurowanego na poziomie szablonu) z najważniejszymi metrykami zebranymi z serwera MySQL.



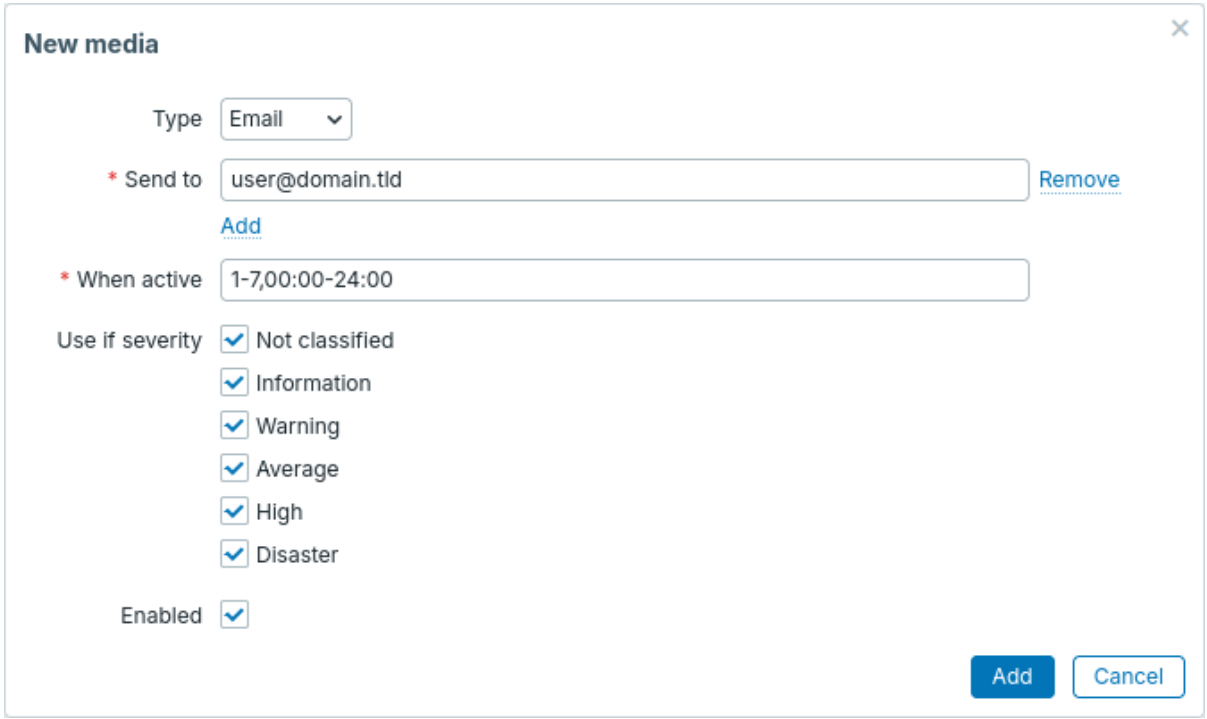
Alternatywnie, w sekcji menu *Monitoring > Hosts* możesz kliknąć *Latest data*, aby wyświetlić wszystkie najnowsze zebrane metryki na liście. Zwróć uwagę, że pozycja *MySQL: Calculated value of innodb_log_file_size* nie powinna zawierać danych, ponieważ wartość zostanie obliczona na podstawie danych z ostatniej godziny.



Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przejdź do zakładki *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem dotyczącym Odbierania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby przetestować swoją konfigurację, możemy zasymulować rzeczywisty problem, aktualizując konfigurację hosta w frontendzie Zabbix.

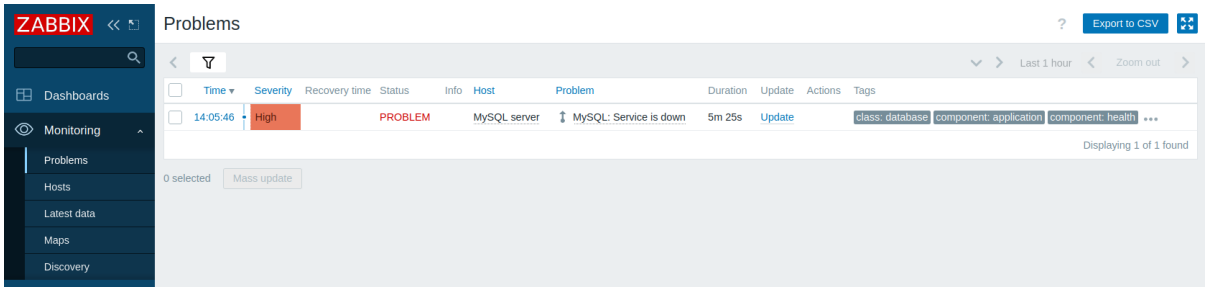
1. Otwórz konfigurację hosta serwera MySQL w Zabbix.

2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok na przykład wcześniej skonfigurowanej wartości makra previously configured {$MYSQL.USER} i ustaw inną nazwę użytkownika MySQL.

4. Kliknij *Update*, aby zaktualizować konfigurację hosta.

5. Po chwili Zabbix wykryje problem ”MySQL: Service is down”, ponieważ nie będzie mógł połączyć się z serwerem MySQL. Problem pojawi się w *Monitoring > Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Zmień wartość makra z powrotem na poprzednią, aby rozwiązać problem i kontynuować monitorowanie serwera MySQL.

Inne podejścia do monitorowania MySQL

Zamiast monitorować serwer MySQL za pomocą agent 2 Zabbixa, można również użyć agenta Zabbixa lub standardu Open Database Connectivity (ODBC). Chociaż zalecane jest użycie agent 2 Zabbixa, mogą istnieć konfiguracje, które nie obsługują agent 2 Zabbixa lub wymagają niestandardowego podejścia.

Kluczowa różnica między agentem Zabbixa a ODBC polega na metodzie zbierania danych - agent Zabbixa jest instalowany bezpośrednio na serwerze MySQL i zbiera dane przy użyciu wbudowanej funkcjonalności, natomiast ODBC opiera się na sterowniku ODBC, aby nawiązać połączenie z serwerem MySQL i pobrać dane za pomocą zapytań SQL.

Chociaż wiele kroków konfiguracji jest podobnych do monitorowania serwera MySQL za pomocą agent 2 Zabbixa, istnieją pewne istotne różnice - aby monitorować serwer MySQL, należy skonfigurować agenta Zabbixa lub ODBC. Poniższe instrukcje przeprowadzą Cię przez te **różnice**.

Monitorowanie MySQL za pomocą agent

Aby monitorować serwer MySQL za pomocą agent, należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

Po pomyślnym zainstalowaniu wymaganych komponentów Zabbix należy utworzyć użytkownika MySQL, jak opisano w sekcji *Tworzenie użytkownika MySQL*.

Po utworzeniu użytkownika MySQL należy skonfigurować agent Zabbix tak, aby mógł nawiązać połączenie z serwerem MySQL i monitorować go. Obejmuje to skonfigurowanie wielu parametrów użytkownika do wykonywania niestandardowych sprawdzeń agent, a także przekazanie agent Zabbix niezbędnych danych uwierzytelniających do łączenia się z serwerem MySQL jako użytkownik ”zbx_monitor” utworzony wcześniej.

Konfigurowanie agenta Zabbix

1. Przejdź do katalogu dodatkowych konfiguracji agenta Zabbix.

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

> **Attention:**
> Katalog dodatkowych konfiguracji agenta Zabbix powinien znajdować się w tym samym katalogu co plik konfiguracyjny agenta Zabbix; domyślna ścieżka zależy od systemu operacyjnego i metody instalacji Zabbix.

Zamiast definiować wszystkie niezbędne parametry użytkownika do monitorowania serwera MySQL w pliku konfiguracyjnym agenta Zabbix, parametry te zostaną zdefiniowane w osobnym pliku w katalogu dodatkowych konfiguracji.

2. Utwórz plik `template_db_mysql.conf` w katalogu dodatkowych konfiguracji agenta Zabbix.

```
vi template_db_mysql.conf
```

3. Skopiuj zawartość pliku `template_db_mysql.conf` (znajdującego się w repozytorium Zabbix) do utworzonego pliku `template_db_mysql.conf` i zapisz plik.

4. Uruchom ponownie agenta Zabbix, aby zaktualizować jego konfigurację.

```
systemctl restart zabbix-agent
```

Po skonfigurowaniu parametrów użytkownika agenta Zabbix możesz przejść do konfigurowania danych uwierzytelniających, które umożliwią agentowi Zabbix dostęp do serwera MySQL.

5. Przejdź do katalogu domowego agenta Zabbix (jeśli nie istnieje w Twoim systemie, musisz go utworzyć; domyślnie: */var/lib/zabbix*).

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

6. Utwórz plik *.my.cnf* w katalogu domowym agenta Zabbix.

```
vi .my.cnf
```

7. Skopiuj poniższą zawartość do pliku *.my.cnf* (zastąp <*password*> hasłem użytkownika ”zbx_monitor”).

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

Skonfiguruj frontend Zabbix i przetestuj swoją konfigurację

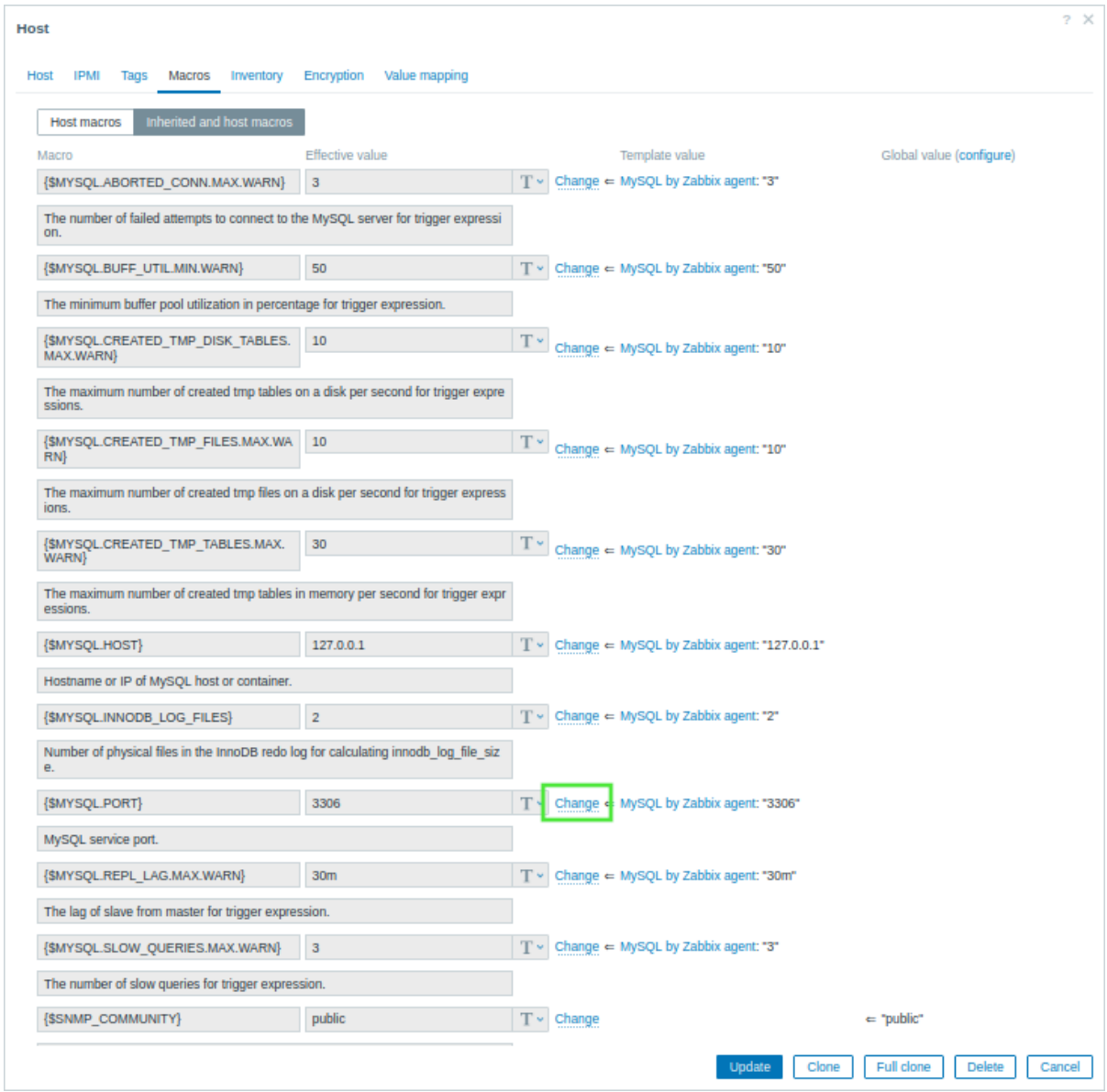
Aby skonfigurować frontend Zabbix, postępuj zgodnie z instrukcjami w sekcji *Skonfiguruj frontend Zabbix*, wprowadzając następujące zmiany:

- W polu *Szablony* wpisz lub wybierz szablon ”MySQL by Zabbix agent”, który zostanie powiązany z hostem.
- Konfigurowanie *Makr* nie jest wymagane.

Po skonfigurowaniu frontendu Zabbix możesz wyświetlić zebrane metryki oraz skonfigurować alerty o problemach.

Aby przetestować konfigurację, postępuj zgodnie z instrukcjami w sekcji *Przetestuj swoją konfigurację*, wprowadzając następujące zmiany:

- W sekcji *Dziedziczone i makra hosta* konfiguracji hosta serwera MySQL kliknij *Zmień* obok wartości makra {$MYSQL.PORT} i ustaw inny port (np. ”6033”).



Monitorowanie MySQL za pomocą ODBC

Aby monitorować serwer MySQL za pomocą ODBC, należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix.

Po pomyślnym zainstalowaniu wymaganych komponentów Zabbix należy utworzyć użytkownika MySQL zgodnie z opisem w sekcji *Utwórz użytkownika MySQL*.

Po utworzeniu użytkownika MySQL należy skonfigurować ODBC. Obejmuje to instalację jednej z najczęściej używanych implementacji otwartego API ODBC - unixODBC - oraz sterownika unixODBC, a także edycję pliku konfiguracji sterownika ODBC.

Konfiguracja ODBC

1. Zainstaluj unixODBC. Zalecanym sposobem instalacji unixODBC jest użycie domyślnych repozytoriów pakietów systemu Linux.

```
apt install unixodbc
```

2. Zainstaluj sterownik bazy danych MariaDB unixODBC. Chociaż używasz bazy danych MySQL, sterownik MariaDB unixODBC jest stosowany ze względu na kwestie zgodności.

```
apt install odbc-mariadb
```

3. Sprawdź lokalizację plików konfiguracyjnych ODBC *odbcinst.ini* i *odbc.ini*.

```
odbcinst -j
```

Wynik wykonania tego polecenia powinien być podobny do poniższego.

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

4. Aby skonfigurować sterownik ODBC do monitorowania bazy danych MySQL, potrzebujesz nazwy sterownika, która znajduje się w pliku *odbcinst.ini*. W poniższym przykładzie pliku *odbcinst.ini* nazwa sterownika to "MariaDB Unicode".

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

5. Skopiuj poniższą zawartość do pliku *odbc.ini* (zastąp <*password*> hasłem użytkownika "zbx_monitor"). W tym przewodniku jako adres serwera MySQL użyto "127.0.0.1" (localhost) do monitorowania serwera MySQL zainstalowanego na tym samym komputerze co sterownik ODBC. Zwróć uwagę na nazwę źródła danych (DSN) "test", która będzie wymagana podczas konfiguracji frontend Zabbix.

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

Skonfiguruj frontend Zabbix i przetestuj swoją konfigurację

Aby skonfigurować frontend Zabbix, postępuj zgodnie z instrukcjami w sekcji *Skonfiguruj frontend Zabbix*, wprowadzając następujące zmiany:

- W polu *Szablony* wpisz lub wybierz szablon "MySQL by ODBC", który zostanie powiązany z hostem.
- Konfigurowanie *Interfejsów* nie jest wymagane.
- Wartość makra {$MYSQL.DSN} w sekcji *Dziedziczone i makra hosta* konfiguracji hosta serwera MySQL powinna być ustawiona na nazwę DSN z pliku *odbc.ini*.

Po skonfigurowaniu frontendu Zabbix możesz wyświetlić zebrane metryki, skonfigurować alerty o problemach oraz przetestować swoją konfigurację.

Zobacz także

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).

- Monitorowanie ODBC - jak skonfigurować ODBC w innych dystrybucjach Linuksa oraz jak rozpocząć monitorowanie dodatkowych metryk związanych z bazą danych za pomocą ODBC.
- Szablon *MySQL by Zabbix agent* - dodatkowe informacje o szablonie *MySQL by Zabbix agent*.
- Szablon *MySQL by Zabbix agent 2* - dodatkowe informacje o szablonie *MySQL by Zabbix agent 2*.
- Szablon *MySQL by ODBC* - dodatkowe informacje o szablonie *MySQL by ODBC*.

Monitoruj VMware za pomocą Zabbix

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania VMware.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania VMware. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Monitorowanie maszyn wirtualnych lub sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

W zależności od środowiska niektóre kroki opisane w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku działającym pod kontrolą Ubuntu.

Zakłada się, że VMware jest już skonfigurowany. Ten przewodnik nie obejmuje konfiguracji VMware.

Konfiguracja serwera Zabbix

Aby monitorować VMware, należy włączyć procesy Zabbix *vmware collector*. Więcej informacji na temat sposobu przeprowadzania monitorowania VMware można znaleźć w sekcji Monitorowanie maszyn wirtualnych.

1. Otwórz plik konfiguracyjny serwera Zabbix.

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

2. Znajdź parametr `StartVMwareCollectors` w pliku konfiguracyjnym serwera Zabbix i ustaw go na wartość 2 lub większą (wartość domyślna to 0).

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

StartVMwareCollectors=2
```

3. Uruchom ponownie serwer Zabbix.

```
systemctl restart zabbix-server
```

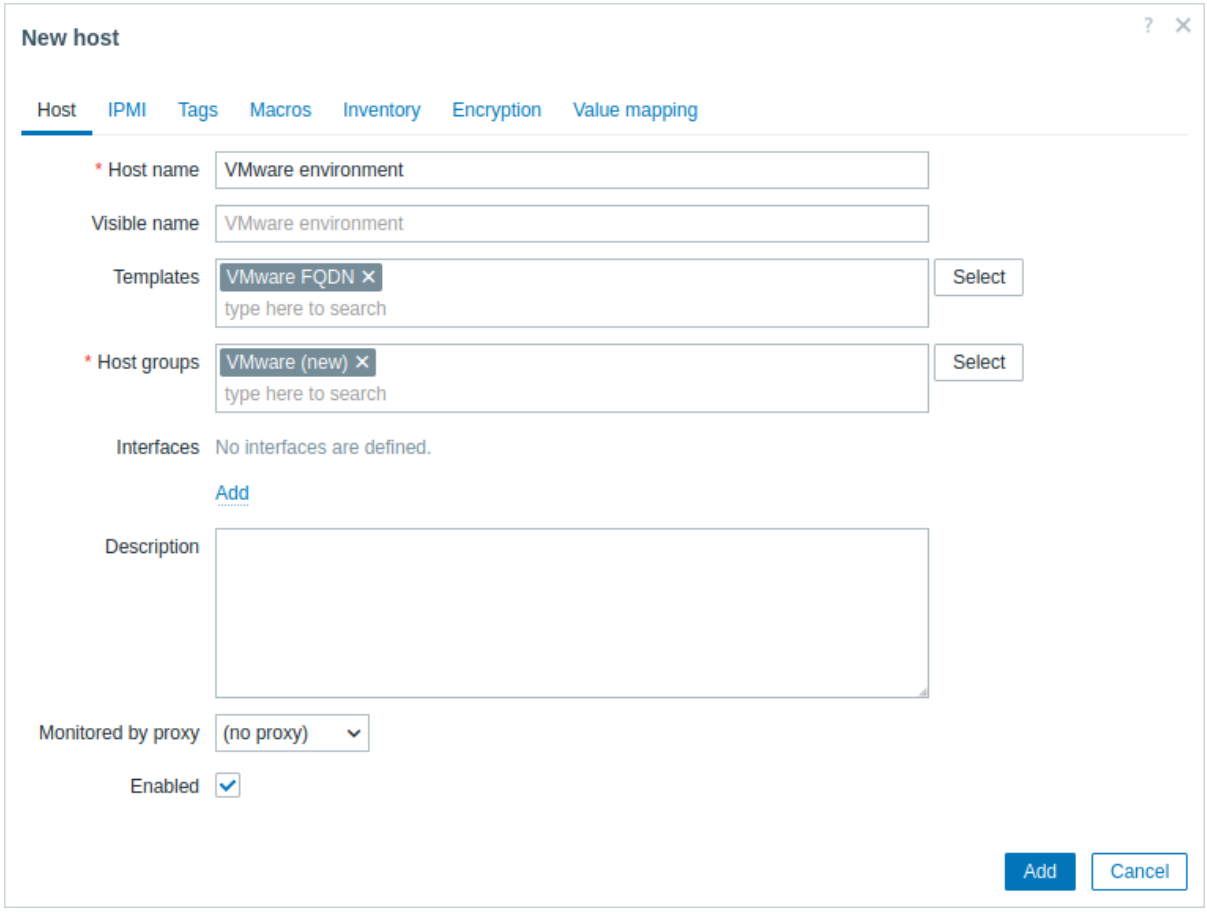
Po uruchomieniu procesów *vmware collector* przejdź do następnego kroku.

Skonfiguruj frontend Zabbix

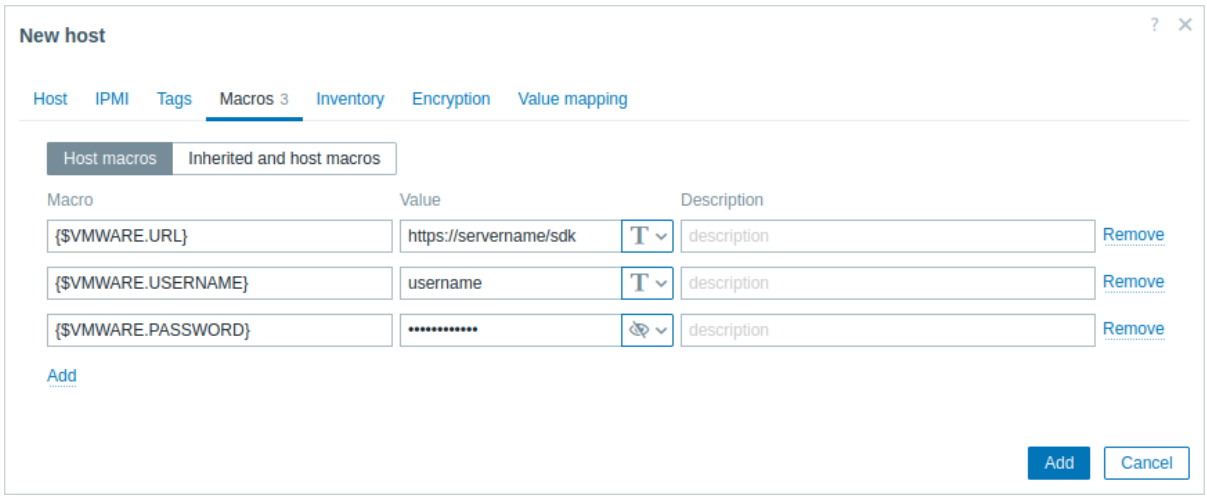
1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie web Zabbix:

- W polu *Host name* wpisz nazwę hosta (na przykład ”VMware environment”).
- W polu *Templates* wpisz lub wybierz szablon ”VMware FQDN” (lub ”VMware”). Więcej informacji o tych szablonach znajdziesz w sekcji Monitorowanie maszyn wirtualnych.
- W polu *Host groups* wpisz lub wybierz grupę hostów (na przykład nową grupę hostów ”VMware”).



- Na karcie *Macros* ustaw następujące makra hosta:
  - {$VMWARE.URL} - adres URL SDK usługi VMware (vCenter lub hiperwizor ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} - nazwa użytkownika usługi VMware
  - {$VMWARE.PASSWORD} - hasło użytkownika {$VMWARE.USERNAME} usługi VMware



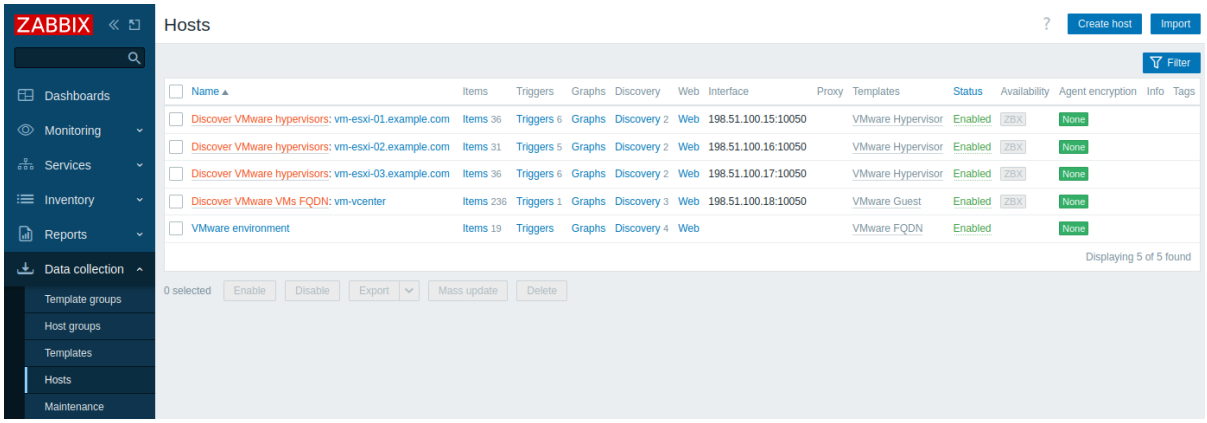
3. Kliknij przycisk *Add*, aby utworzyć host. Ten host będzie reprezentował Twoje środowisko VMware.

Wyświetlanie zebranych metryk

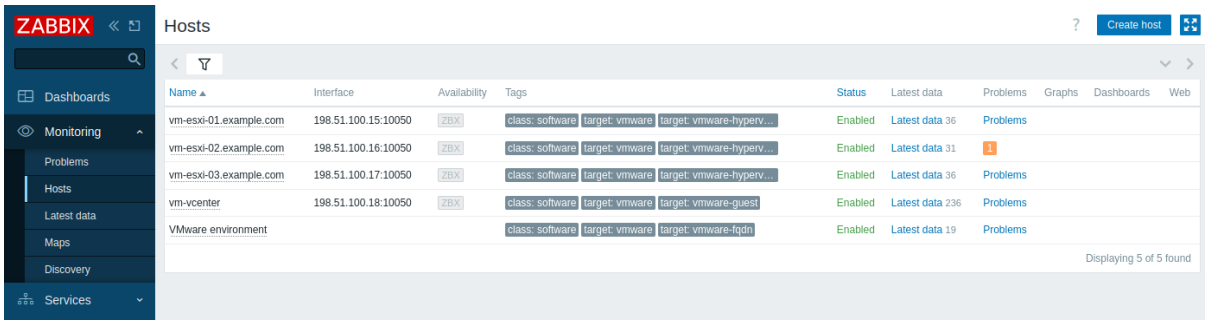
Gratulacje! W tym momencie Zabbix już monitoruje Twoje środowisko VMware.

W zależności od konfiguracji środowiska VMware, Zabbix może wykryć, a następnie utworzyć hosty dla wykrytych encji. Należy pamiętać, że wykrywanie i tworzenie hostów może również zostać wykonane ręcznie, jeśli zajdzie taka potrzeba.

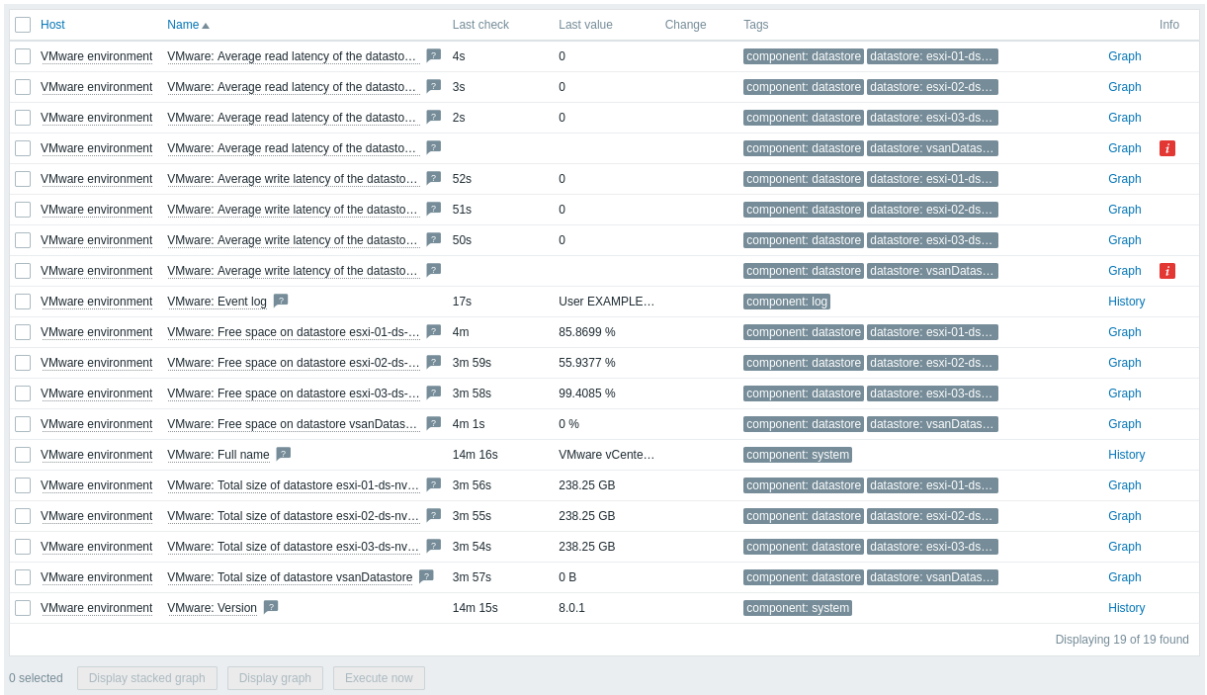
Aby wyświetlić utworzone hosty, przejdź do sekcji menu *Zbieranie danych > Hosty*.



Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring > Hosty* i kliknij *Najnowsze dane* obok utworzonego hosta "VMware environment" lub jednego z hostów utworzonych dla wykrytych encji.



Ta akcja otworzy listę wszystkich najnowszych metryk zebranych z wybranego hosta.

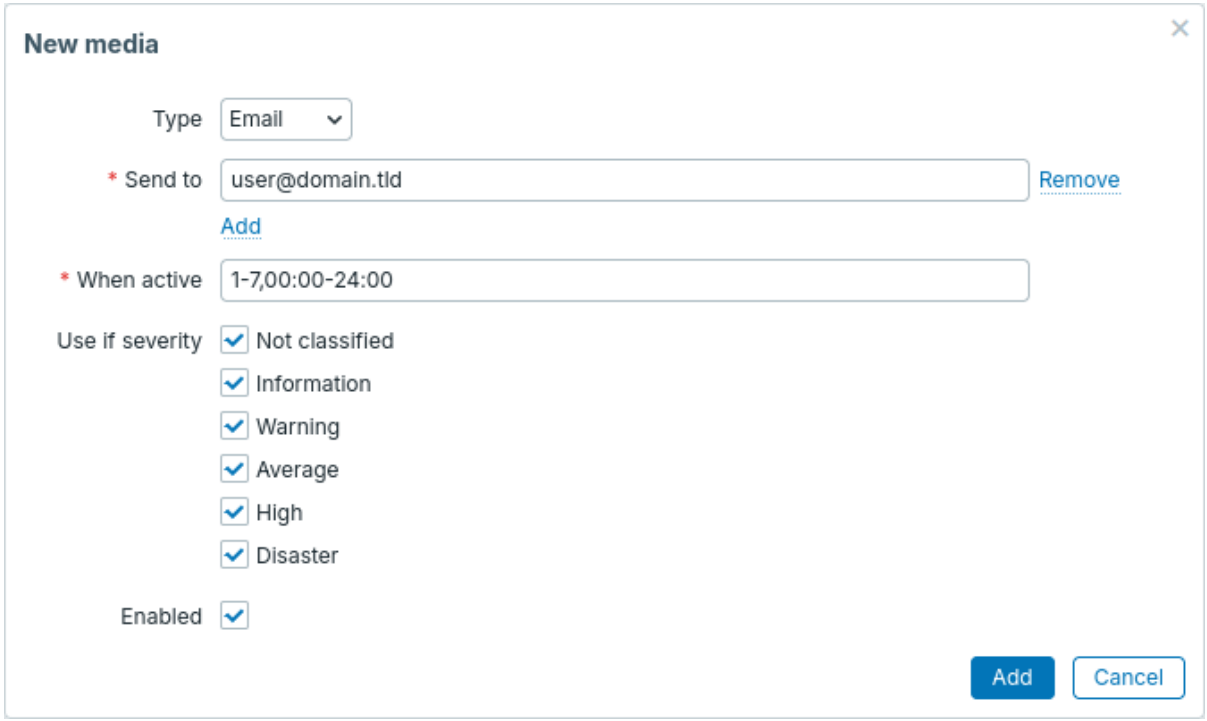


Należy pamiętać, że niektóre pozycje nie mają danych i mają stan *Niewspierane*. Dzieje się tak, ponieważ Zabbix nie może znaleźć prawidłowych liczników wydajności dla konkretnego datastore, ponieważ nie jest on włączony w monitorowanym środowisku VMware.

Skonfiguruj alerty o problemach

Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przejdź do zakładki *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem dotyczącym Odbierania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Zobacz także

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Monitorowanie maszyn wirtualnych - dodatkowe informacje o monitorowaniu VMware (proces zbierania danych, opcje konfiguracji serwera, wskazówki dotyczące rozwiązywania problemów itp.).
- Klucze pozycji monitorowania VMware - pełna lista metryk VMware, które można monitorować za pomocą Zabbixa.
- Szablon *VMware* - dodatkowe informacje o szablonie *VMware*.
- Szablon *VMware FQDN* - dodatkowe informacje o szablonie *VMware FQDN*.

Monitorowanie ruchu sieciowego za pomocą Zabbix

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania ruchu sieciowego za pomocą Zabbix.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania ruchu sieciowego. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Configuration w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego. Należy pamiętać, że agent Zabbix powinien zostać zainstalowany na maszynie, która wymaga monitorowania ruchu sieciowego. Może to być zarówno ten sam host, na którym zainstalowano serwer Zabbix, jak i inny host.

Ten przewodnik zawiera instrukcje dotyczące konfiguracji monitorowania ruchu sieciowego interfejsu *eth0* na oddzielnej maszynie o nazwie *Remote host*.

Konfiguracja Zabbixa do monitorowania

Agent Zabbixa może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie). Więcej informacji można znaleźć w sekcji Pasywne i aktywne sprawdzenia agenta. W tym przewodniku zostanie opisane monitorowanie za pomocą sprawdzeń pasywnych.

Konfigurowanie agent

1. Otwórz plik konfiguracyjny agent na maszynie, na której zainstalowano agent:

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

2. Dodaj adres IP lub nazwę DNS serwera Zabbix do parametru *Server*. Na przykład:

```
Server=192.0.2.22
```

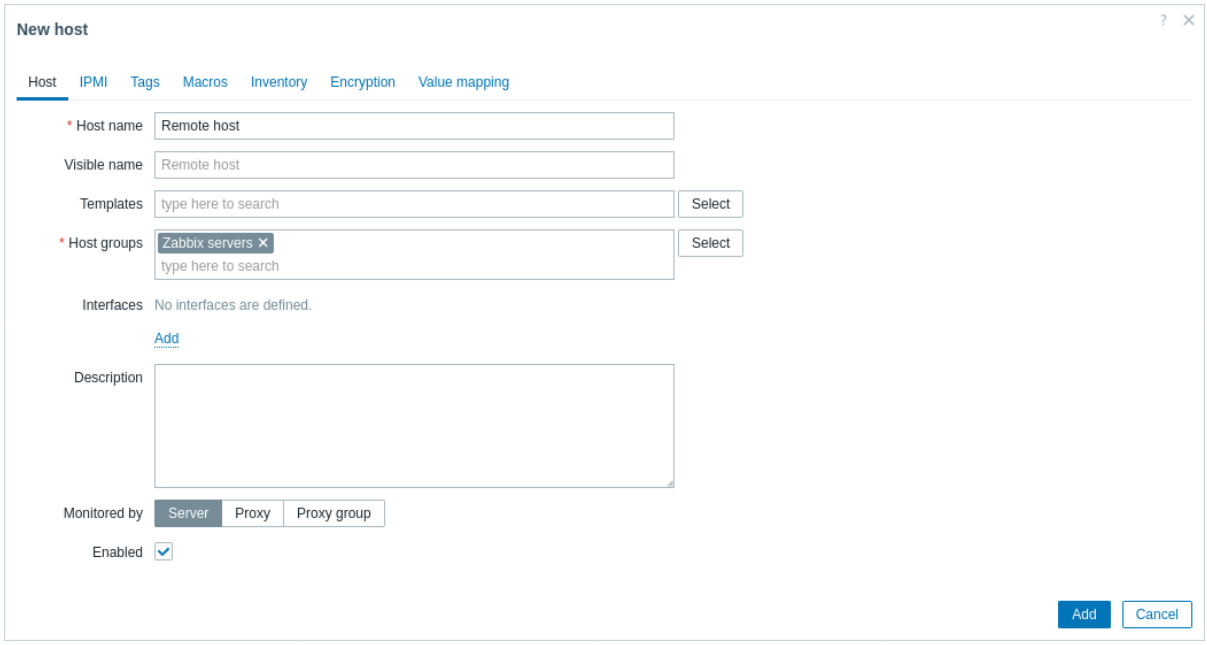
3. Uruchom ponownie agent Zabbix:

```
systemctl restart zabbix-agent
```

Frontend Zabbix

1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie web Zabbix, podając adres IP lub nazwę DNS komputera, na którym zainstalowano agent.

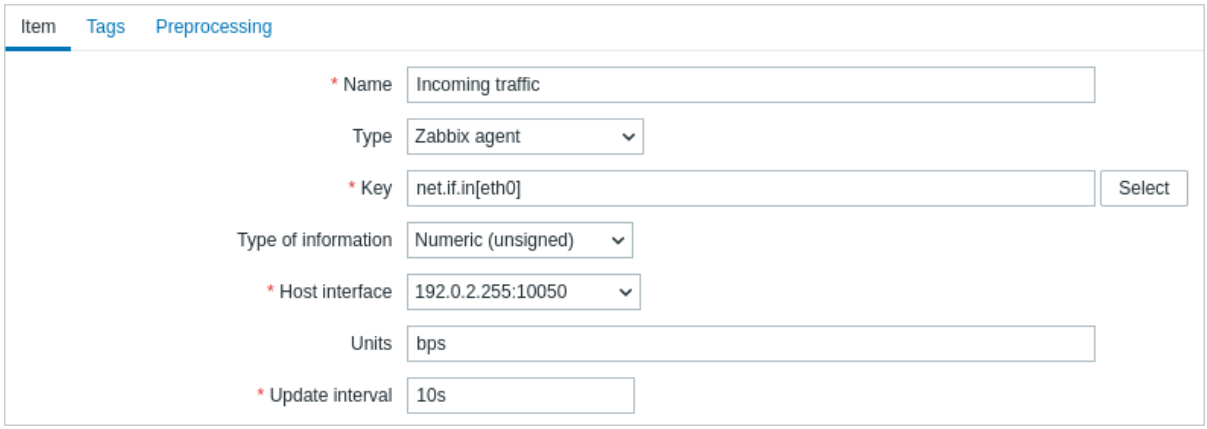


Tworzenie pozycji

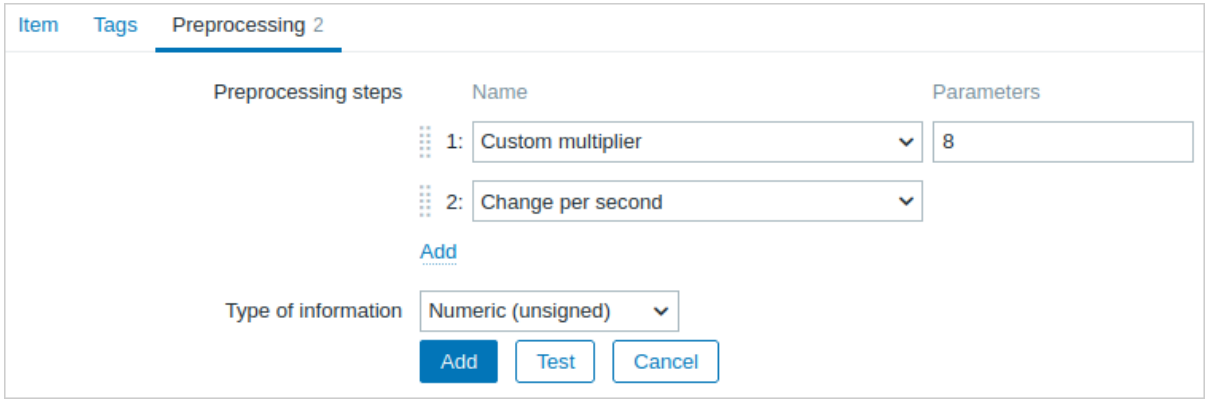
Postępuj zgodnie z instrukcjami dotyczącymi tworzenia pozycji, aby dodać pozycje do monitorowania ruchu, a mianowicie:

- Ruch przychodzący
- Ruch wychodzący
- Całkowity ruch

Prosta konfiguracja pozycji monitorującej ruch przychodzący wyglądałaby następująco:



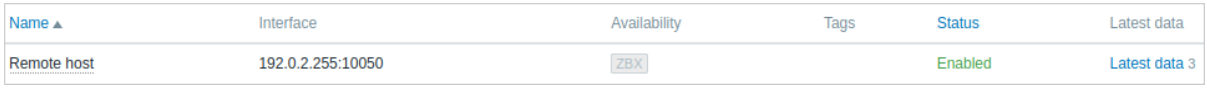
Aby zebrane dane nadawały się do praktycznego wykorzystania, podczas tworzenia pozycji można ustawić kilka kroków preprocessing. W tym przypadku mogą to być mnożenie przez 8 (w celu przeliczenia bajtów na bity) oraz prezentacja jako zmiana na sekundę.



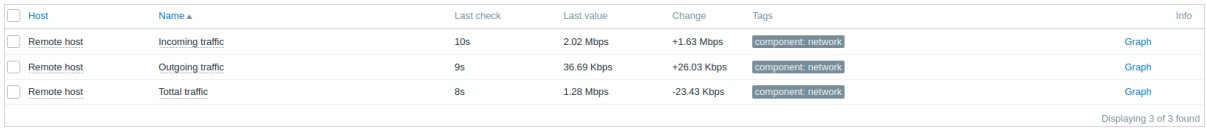
Wyświetlanie zebranych danych

Gratulacje! W tym momencie Zabbix już monitoruje ruch sieciowy.

Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring > Hosts* i kliknij *Latest data* w wierszu hosta.

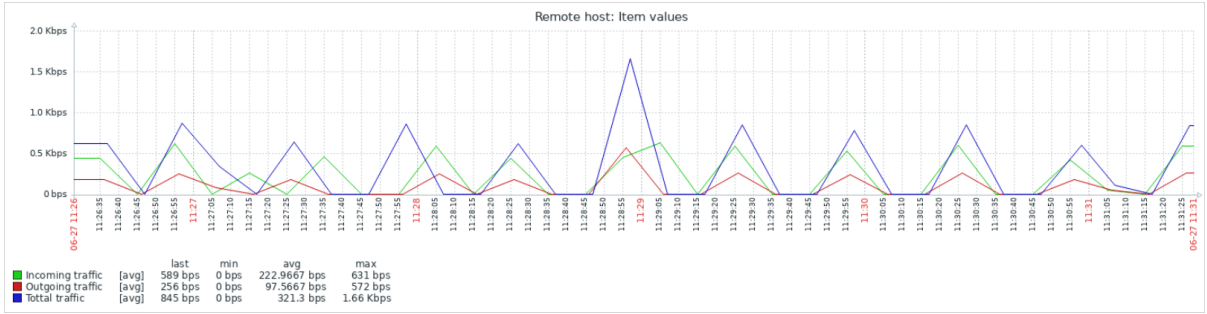


Zobaczysz dane dotyczące ruchu.



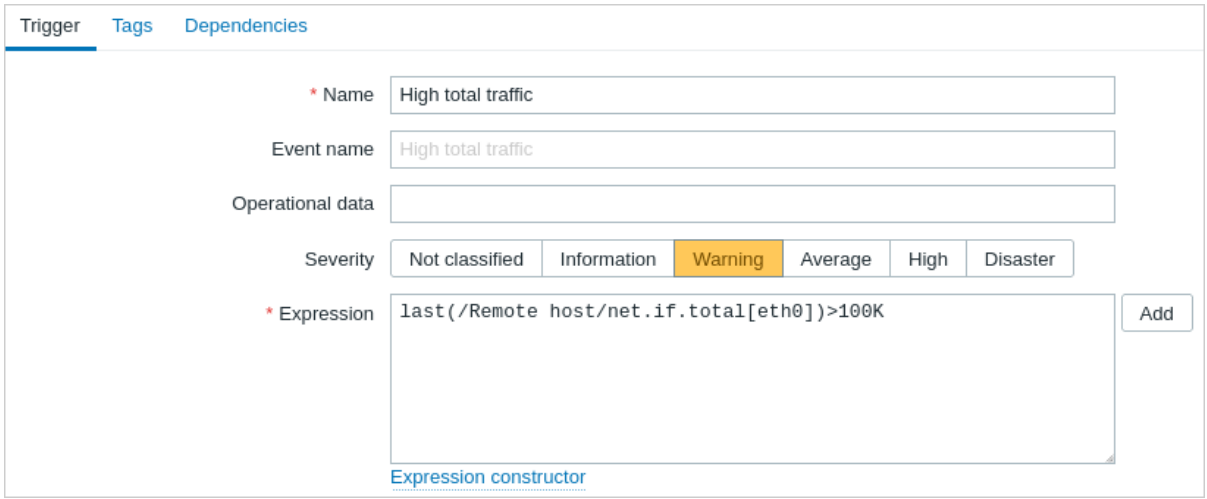
Wyświetlanie wykresów

Zebrane dane można wyświetlać jako wykresy. Aby je zobaczyć, w sekcji *Najnowsze dane* kliknij *Wykres* w wierszu pozycji lub zaznacz wymagane pozycje i kliknij *Wyświetl wykres* poniżej.

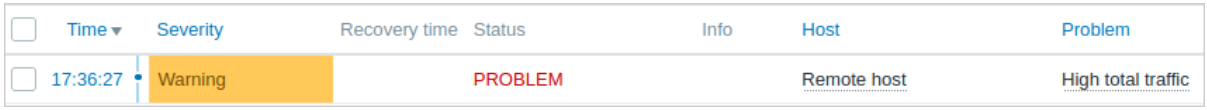


Konfigurowanie wyzwalaczy

Możesz ustawić wyzwalacze, aby wykrywać nietypowy ruch sieciowy. Zobacz instrukcje dotyczące konfigurowania wyzwalacza i dodaj wyzwalacz sygnalizujący, że całkowity ruch jest zbyt wysoki, na przykład:



Teraz spraw, aby ruch przekroczył próg ustawiony w wyrażeniu wyzwalacza, a następnie przejdź do *Monitoring > Problems*, aby sprawdzić, czy problem jest tam widoczny.



Time ▾	Severity	Recovery time	Status	Info	Host	Problem
17:36:27	Warning		PROBLEM		Remote host	High total traffic

Skonfiguruj alerty o problemach

Istnieje kilka sposobów otrzymywania powiadomień o problemie. E-mail jest najpopularniejszą opcją, dlatego postępuj zgodnie z instrukcjami dotyczącymi konfiguracji powiadomienia o problemie za pośrednictwem poczty e-mail. Możesz również wybrać inne typy mediów, które będą używane do dostarczania powiadomień.

Zobacz także:

- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Potwierdzenie problemu - jak wskazać, że problem jest znany, dodawać komentarze dotyczące jego rozwiązania, wstrzymywać lub zamykać problem.
- Monitorowanie systemu Linux za pomocą agenta Zabbix - jak rozpocząć podstawowe monitorowanie najważniejszych pozycji poprzez powiązanie wstępnie skonfigurowanego szablonu.

Monitorowanie ruchu sieciowego za pomocą aktywnych kontroli

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania ruchu sieciowego za pomocą Zabbix przy użyciu aktywnych kontroli.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania ruchu sieciowego przy użyciu aktywnych kontroli. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego. Należy pamiętać, że agent Zabbix powinien zostać zainstalowany na maszynie, która wymaga monitorowania ruchu sieciowego. Może to być ten sam host, na którym zainstalowano serwer Zabbix, albo inny host.

Ten przewodnik zawiera instrukcje dotyczące konfiguracji monitorowania ruchu sieciowego interfejsu *eth0* na oddzielnej maszynie o nazwie *Remote host*.

Skonfiguruj Zabbix do monitorowania

agent Zabbix może zbierać metryki w trybie aktywnym lub pasywnym (jednocześnie). Więcej informacji znajdziesz w sekcji Pasywne i aktywne sprawdzenia agenta. W tym przewodniku zostanie opisane monitorowanie za pomocą **sprawdzeń aktywnych**.

Konfigurowanie Zabbix agent

1. Otwórz plik konfiguracyjny agenta na maszynie, na której jest zainstalowany agent.

Jeśli używasz Zabbix agent:

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

Jeśli używasz Zabbix agent 2:

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

2. Dodaj adres IP lub nazwę DNS (oraz opcjonalnie port) serwera Zabbix do parametru ServerActive. Na przykład:

```
ServerActive=192.0.2.1:10051
```

Zabbix agent użyje tego adresu, aby połączyć się z portem trapper serwera Zabbix (domyślnie: 10051) i zażądać danych konfiguracyjnych dotyczących aktywnych kontroli.

3. Zdefiniuj parametr Hostname, który musi odpowiadać nazwie hosta zdefiniowanej w frontend Zabbix. W naszym przykładzie jest to:

```
Hostname=Remote host
```

Wartość Hostname musi się zgadzać, ponieważ w przypadku aktywnych kontroli Zabbix agent używa jej do pobrania prawidłowej konfiguracji hosta z serwera. W szczególności agent inicjuje połączenie z serwerem i identyfikuje się za pomocą wartości Hostname. Następnie serwer udostępnia konfigurację monitorowania dla tego hosta. Jeśli wartości te się różnią, agent nie otrzyma odpowiedniej konfiguracji, co spowoduje brak metryk lub problemy z monitorowaniem.

4. Uruchom ponownie Zabbix agent.

Jeśli używasz Zabbix agent:

```
systemctl restart zabbix-agent
```

Jeśli używasz Zabbix agent 2:

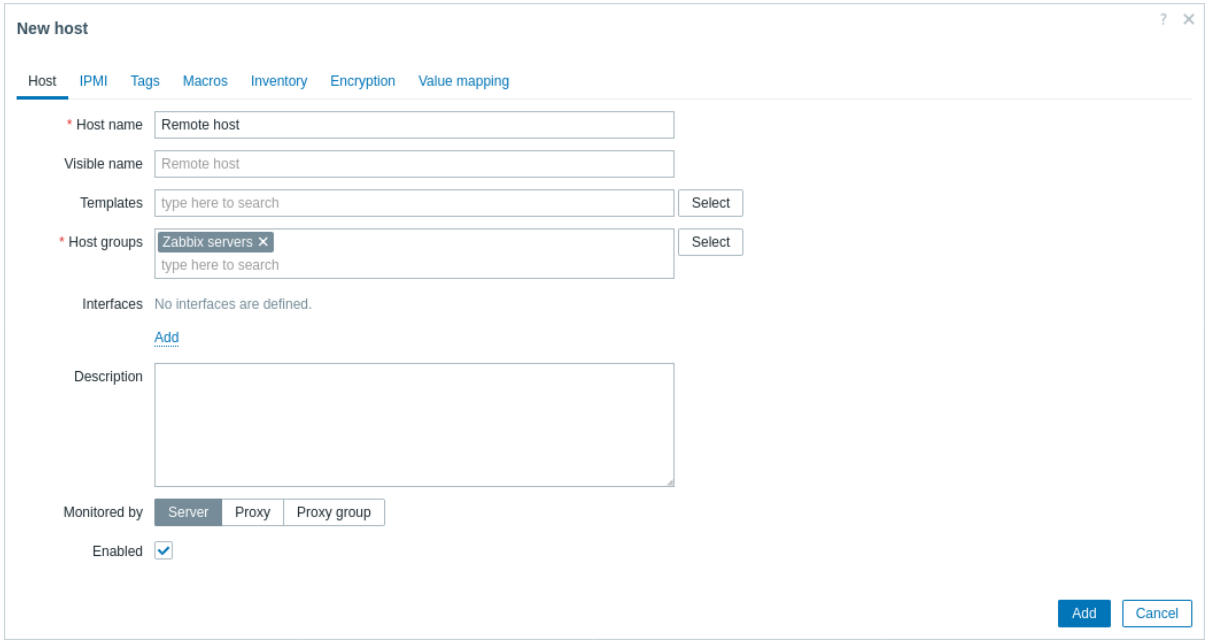
```
systemctl restart zabbix-agent2
```

Frontend Zabbix

1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie web Zabbix.

- W polu *Host name* wpisz nazwę hosta (np. ”Remote host”), która odpowiada wartości parametru Hostname zdefiniowanej wcześniej w pliku konfiguracyjnym agent.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. ”Zabbix servers”).
- Pozostaw *Interfaces* niezdefiniowane; interfejs nie jest potrzebny dla aktywnych kontroli, ponieważ agent inicjuje połączenie z serwerem zamiast nasłuchiwać na połączenie z serwera.



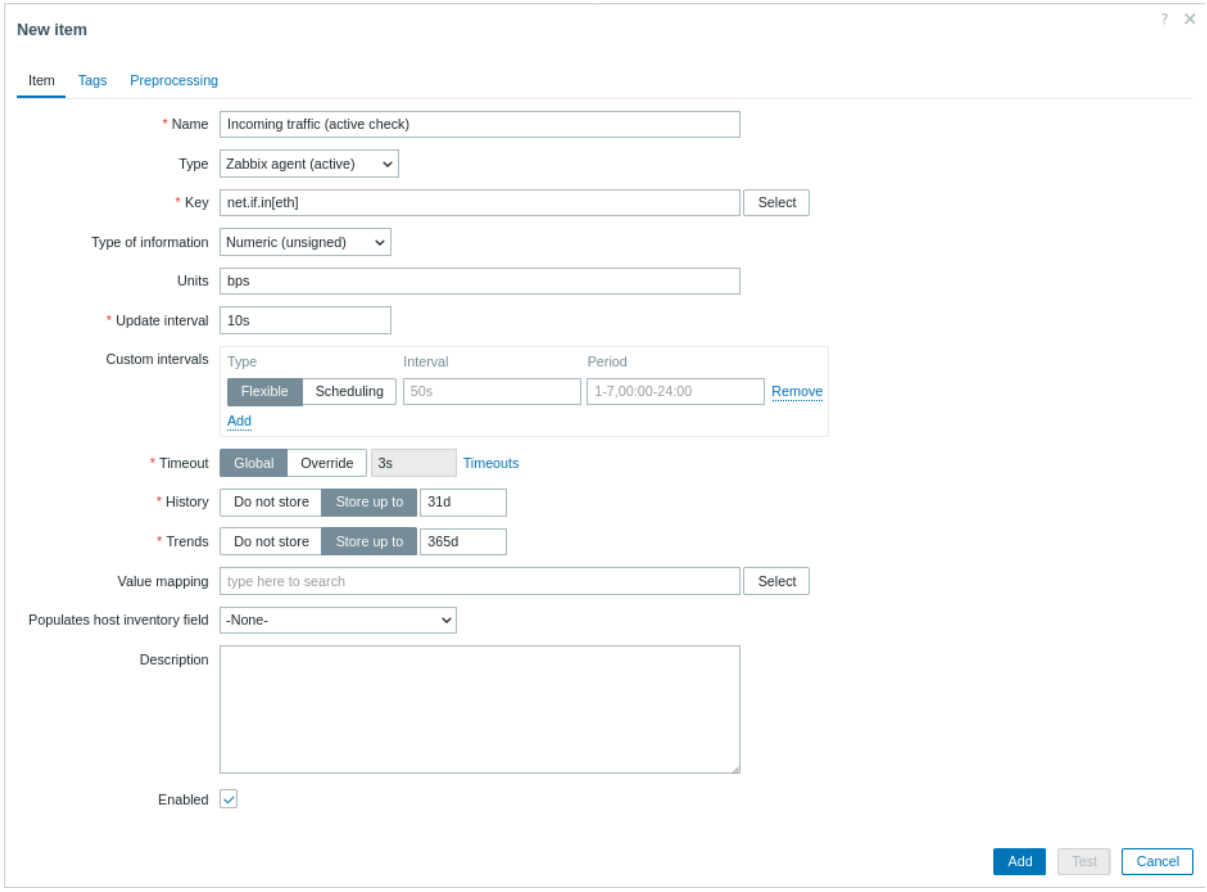
3. Kliknij *Add*, aby dodać host. Ten host będzie reprezentował monitorowaną maszynę Linux.

Tworzenie pozycji

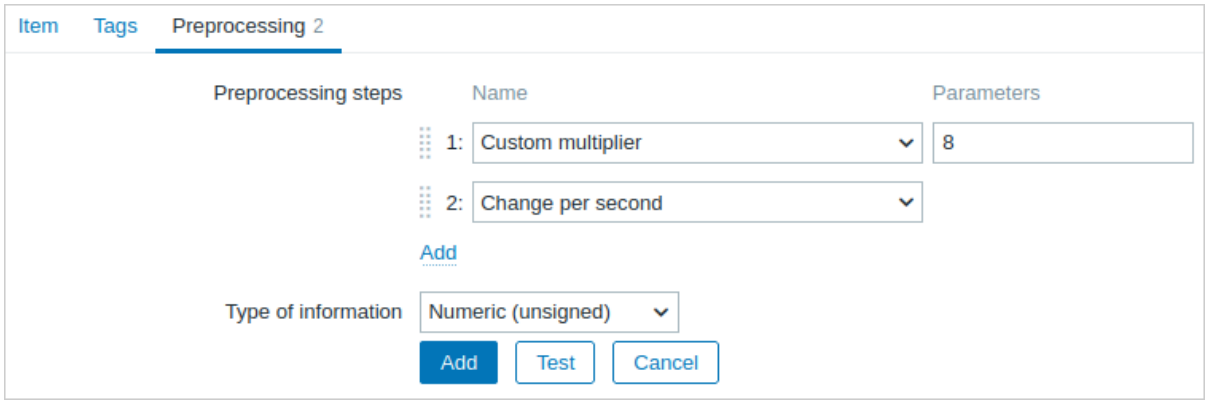
Postępuj zgodnie z instrukcjami dotyczącymi tworzenia pozycji, aby dodać pozycje do monitorowania ruchu, a mianowicie:

- Ruch przychodzący
- Ruch wychodzący
- Całkowity ruch

Prosta konfiguracja pozycji monitorującej ruch przychodzący z aktywnym sprawdzaniem wyglądałaby następująco:



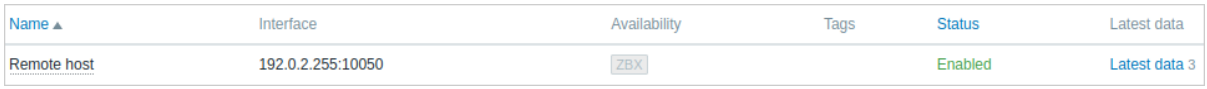
Aby zebrane dane nadawały się do praktycznego wykorzystania, podczas tworzenia pozycji można ustawić pewne kroki preprocessing. W tym przypadku mogą to być mnożenie przez 8 (w celu przeliczenia bajtów na bity) oraz prezentacja jako zmiana na sekundę.



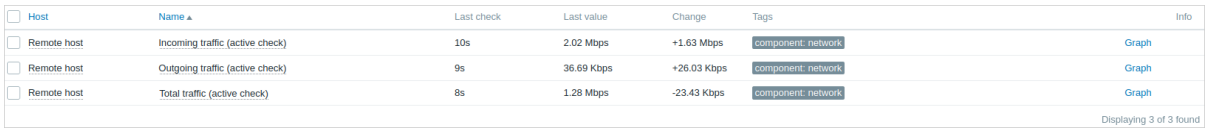
Wyświetlanie zebranych danych

Gratulacje! W tym momencie Zabbix już monitoruje ruch w Twojej sieci.

Aby wyświetlić zebrane metryki, otwórz sekcję menu *Monitoring > Hosts* i kliknij *Latest data* w wierszu hosta.



Zobaczysz dane o ruchu.



Zobacz także:

- Wyświetlanie wykresów - jak wyświetlać zebrane dane w postaci wykresów.
- Konfiguracja wyzwalaczy - jak skonfigurować wyzwalacze do wykrywania nietypowego ruchu sieciowego.
- Konfiguracja alertów o problemach - jak skonfigurować powiadomienia o sytuacjach problemowych.

- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Potwierdzanie problemów - jak wskazać, że problem jest znany, dodawać komentarze dotyczące jego rozwiązania, wstrzymywać lub zamykać problem.
- Monitorowanie systemu Linux za pomocą agenta Zabbixa - jak rozpocząć podstawowe monitorowanie najważniejszych pozycji poprzez powiązanie wstępnie skonfigurowanego szablonu.
- Instalacja z pakietów - jak instalować komponenty Zabbixa przy użyciu oficjalnych pakietów RPM i DEB dla różnych dystrybucji Linuxa, zapewniając dostęp do najnowszych funkcji i poprawek błędów.

Monitorowanie witryn internetowych za pomocą pozycji przeglądarki

Wprowadzenie

Ta strona przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania witryn internetowych za pomocą pozycji Browser.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania witryn internetowych za pomocą pozycji Browser. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz stronę Browser items lub sekcję Configuration w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix oraz frontend Zabbix zgodnie z instrukcjami dla używanego systemu operacyjnego.

W zależności od środowiska niektóre kroki opisane w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku działającym pod kontrolą Ubuntu.

Konfiguracja WebDriver

Pozycje przeglądarki wymagają frameworka automatyzacji (Selenium Server albo zwykłego WebDrivera, na przykład ChromeDriver) jako punktu końcowego testów webowych, który steruje przeglądarką i wchodzi z nią w interakcję, wykonując polecenia testowe, takie jak klikanie przycisków lub wprowadzanie tekstu. Jako przykład w tym przewodniku zostanie użyty Selenium Server z Chrome w kontenerze Docker.

Zakłada się, że Docker jest już skonfigurowany. Ten przewodnik nie obejmuje konfiguracji Dockera. Instrukcje instalacji znajdują się w Install Docker Engine on Ubuntu.

1. Uruchom Selenium Server z Chrome w kontenerze Docker z następującymi opcjami:

- **docker run --name browser** - uruchamia nowy kontener Docker o nazwie "browser";
- **-p 4444:4444** - mapuje port 4444 na komputerze hosta na port 4444 w kontenerze (jest to port używany przez Selenium Server do przyjmowania poleceń);
- **-p 7900:7900** - mapuje port 7900 na komputerze hosta na port 7900 w kontenerze (jest to port używany przez serwer Virtual Network Computing (VNC), umożliwiający zdalne wyświetlanie interfejsu GUI przeglądarki; wymaga klienta VNC);
- **--shm-size="2g"** - przydziela kontenerowi 2 GB pamięci współdzielonej (jest to ważne, aby Chrome działał poprawnie, ponieważ może wymagać znacznej ilości pamięci współdzielonej, aby uniknąć awarii);
- **-d** - uruchamia kontener w trybie odłączonym, co oznacza, że będzie działał w tle;
- **selenium/standalone-chrome:latest** - określa obraz Docker, którego należy użyć; w tym przypadku jest to najnowsza wersja Selenium Server with Chrome.

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

2. Upewnij się, że kontener Docker `browser` jest uruchomiony i dostępny.

- Pobierz adres IP kontenera (w tym przykładzie: 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
#####     ...
```

- Przetestuj połączenie z kontenerem za pomocą Ncat:

```
nc -zv 192.0.2.1 4444

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

- Pobierz zawartość strony internetowej z Selenium Server za pomocą 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:
W razie problemów zapoznaj się z dokumentacją Dockera.

Konfiguracja serwera Zabbix

Pozycje przeglądarki są wykonywane i przetwarzane przez procesy Zabbix typu *browser poller*, które należy włączyć, dostosowując parametr konfiguracji serwera StartBrowserPollers. Dodatkowo parametr WebDriverURL powinien wskazywać wcześniej skonfigurowany endpoint testów internetowych.

Domyślnie parametr StartBrowserPollers ma wartość 1, dlatego wystarczy określić endpoint testów internetowych.

1. Otwórz plik konfiguracji serwera Zabbix:

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

2. Znajdź i ustaw parametr WebDriverURL w pliku konfiguracji serwera 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. Uruchom ponownie serwer Zabbix.

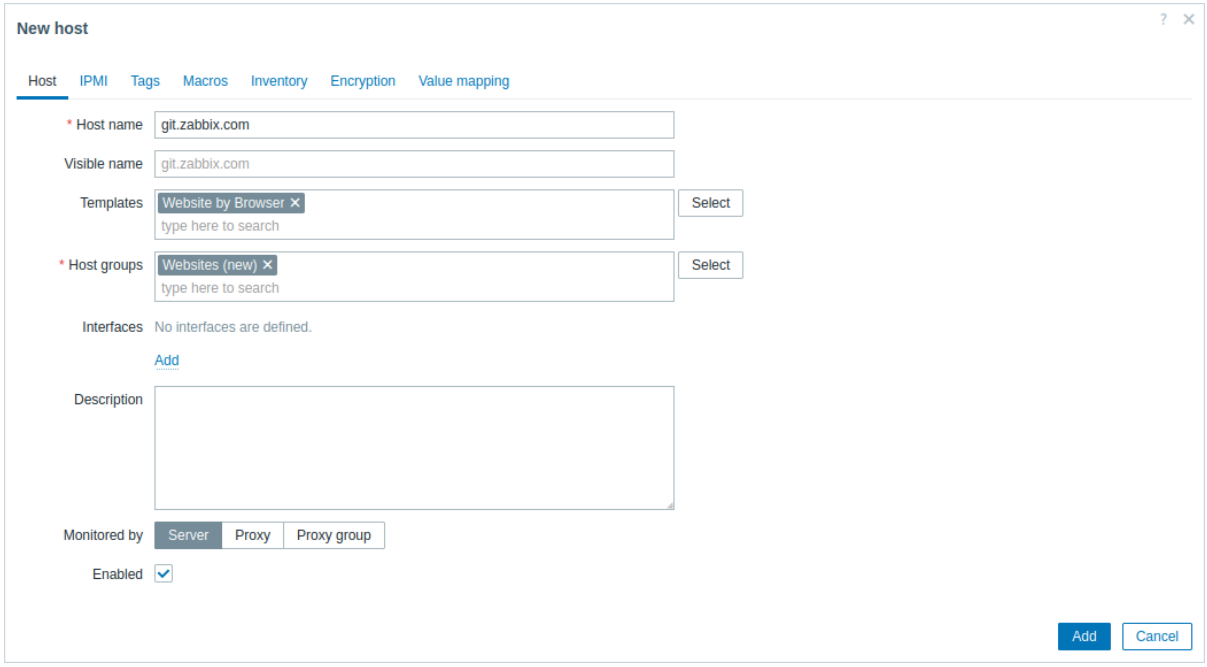
```
systemctl restart zabbix-server
```

Skonfiguruj frontend Zabbix

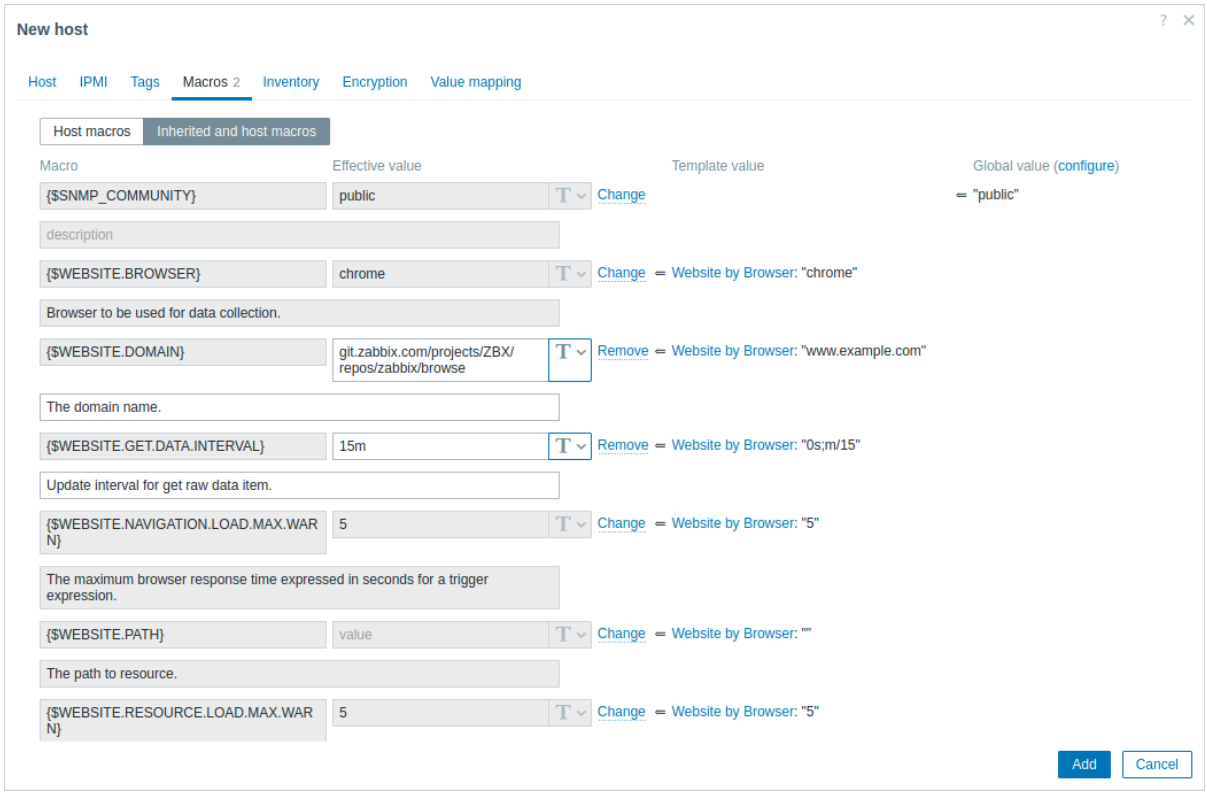
1. Zaloguj się do frontend Zabbix.

2. Utwórz host w interfejsie webowym Zabbix:

- W polu *Host name* wpisz nazwę hosta (na przykład ”git.zabbix.com”).
- W polu *Templates* wpisz lub wybierz szablon ”Website by Browser”. Więcej informacji o tym szablonie znajdziesz tutaj: Website by Browser.
- W polu *Host groups* wpisz lub wybierz grupę hostów (na przykład nową grupę hostów ”Websites”).



- Na karcie *Macros* przełącz się na *Inherited and host macros*, wyszukaj następujące makra i kliknij *Change* obok wartości makra, aby ją zaktualizować:
  - {$WEBSITE.DOMAIN} - Nazwa domeny (na przykład git.zabbix.com/projects/ZBX/repos/zabbix/browse)
  - {$WEBSITE.GET.DATA.INTERVAL} - Interwał aktualizacji danych pozycji (na przykład 15m)

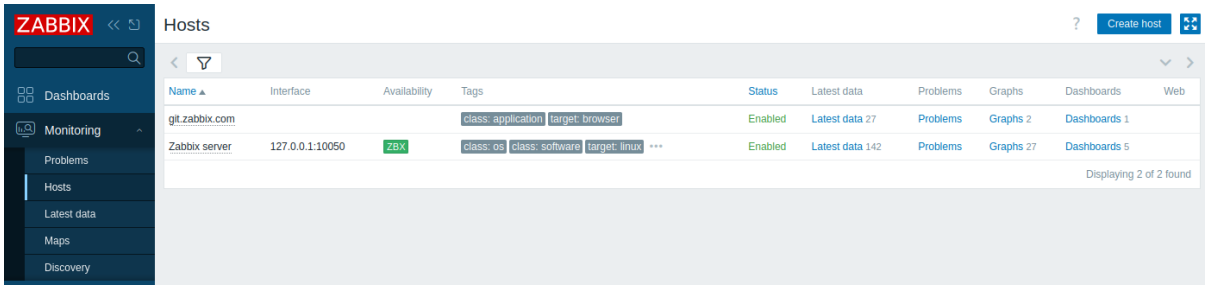


3. Kliknij przycisk *Add*, aby utworzyć host. Ten host będzie reprezentował witrynę, którą chcesz monitorować.

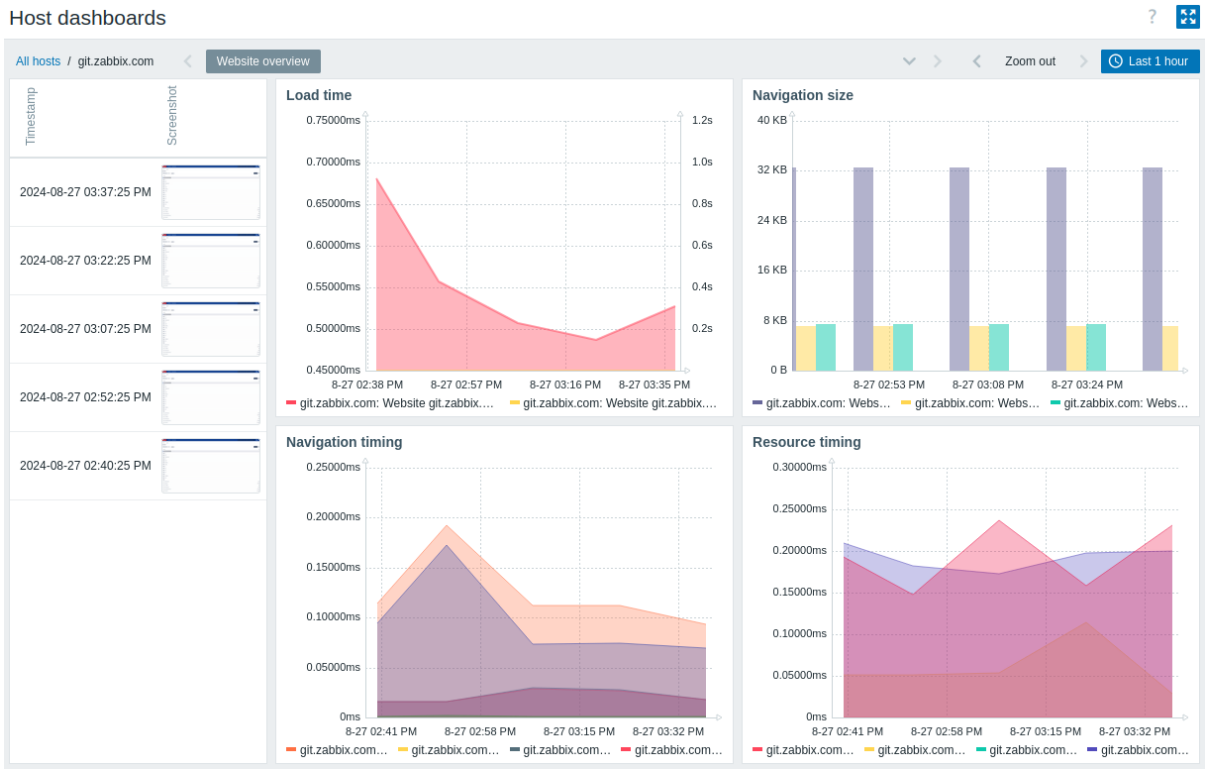
Wyświetlanie zebranych metryk

Gratulacje! W tym momencie Zabbix już monitoruje wskazaną przez Ciebie witrynę.

Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring > Hosts* i kliknij *Dashboards* obok hosta.



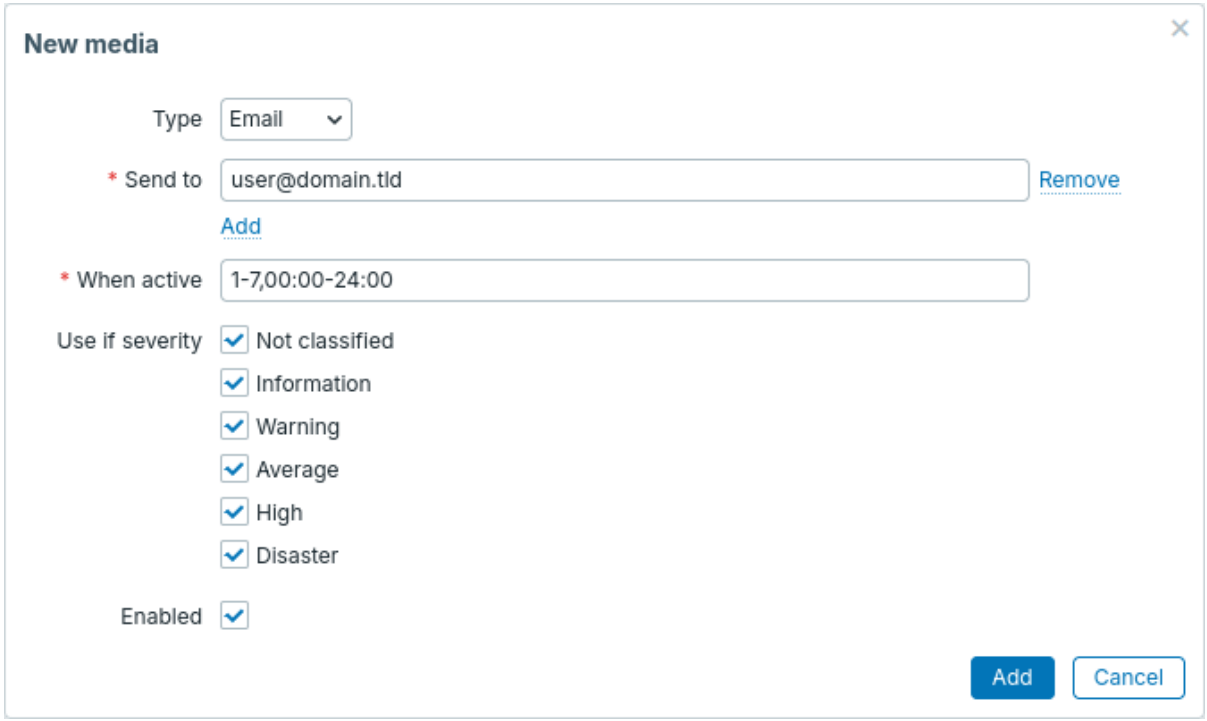
Ta akcja przeniesie Cię do dashboardu hosta (skonfigurowanego na poziomie szablonu) z najważniejszymi metrykami zebranymi z witryny.



Konfiguracja alertów o problemach

Zabbix może powiadamiać o problemach z infrastrukturą za pomocą różnych metod. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem dotyczącym Odbierania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby przetestować swoją konfigurację, możemy zasymulować rzeczywisty problem, aktualizując konfigurację hosta we frontendzie Zabbix.

1. Otwórz konfigurację hosta swojej witryny w Zabbix.

2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok na przykład wcześniej skonfigurowanej wartości makra {$WEBSITE.DOMAIN} i ustaw nieprawidłową nazwę domeny (na przykład /git.zabbix.com/projects/ZBX/repos/zabbix/browse).

4. Kliknij *Update*, aby zaktualizować konfigurację hosta.

5. Po kilku chwilach Zabbix wykryje problem "Failed to get JSON of the requested website", ponieważ nie będzie w stanie połączyć się z określoną witryną. Problem pojawi się w *Monitoring > Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Przywróć poprzednią wartość makra, aby rozwiązać problem i kontynuować monitorowanie witryny.

Zobacz także

- Tworzenie pozycji - jak rozpocząć monitorowanie dodatkowych metryk.
- Eskalacje problemów - jak tworzyć wieloetapowe scenariusze alertów (np. najpierw wysłać wiadomość do administratora systemu, a następnie, jeśli problem nie zostanie rozwiązany w ciągu 45 minut, wysłać wiadomość do kierownika centrum danych).
- Pozycje przeglądarki - jak skonfigurować pozycje przeglądarki.
- Szablon Website by Browser - dodatkowe informacje o szablonie *Website by Browser*.

Monitorowanie certyfikatów witryn internetowych za pomocą agent 2 Zabbix (pasywny)

Wprowadzenie

Ten przewodnik zawiera kompleksowy przegląd sposobu konfiguracji i monitorowania certyfikatów SSL/TLS przy użyciu klucza `web.certificate.get` w agent 2 Zabbix. Został on opracowany, aby uprościć monitorowanie certyfikatów dla jednej lub wielu witryn internetowych, umożliwiając administratorom szybkie wykrywanie potencjalnych problemów, takich jak wygasłe lub nieprawidłowe certyfikaty.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbix i zawiera minimalny zestaw kroków wymaganych do włączenia podstawowego monitorowania certyfikatów witryn internetowych. Jeśli szukasz zaawansowanych opcji dostosowywania lub potrzebujesz bardziej rozbudowanej konfiguracji, zobacz sekcję Konfiguracja w podręczniku Zabbix.

Wymagania wstępne

Przed kontynuowaniem tego przewodnika należy pobrać i zainstalować serwer Zabbix, frontend Zabbix oraz agent Zabbix 2 zgodnie z instrukcjami dla używanego systemu operacyjnego. W tym samouczku przyjęto, że zarówno serwer Zabbix, jak i agent są zainstalowane na tym samym komputerze; dlatego w konfiguracji użyto `127.0.0.1`.

Konfiguracja agenta Zabbix 2

1. Otwórz plik konfiguracyjny agenta Zabbix:

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

2. Ustaw parametr Server na 127.0.0.1, ponieważ agent i serwer działają na tej samej maszynie:

```
Server=127.0.0.1
```

3. Zapisz plik i uruchom ponownie usługę agenta Zabbix 2:

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

4. Po skonfigurowaniu agenta Zabbix 2 przetestuj jego dostępność za pomocą polecenia:

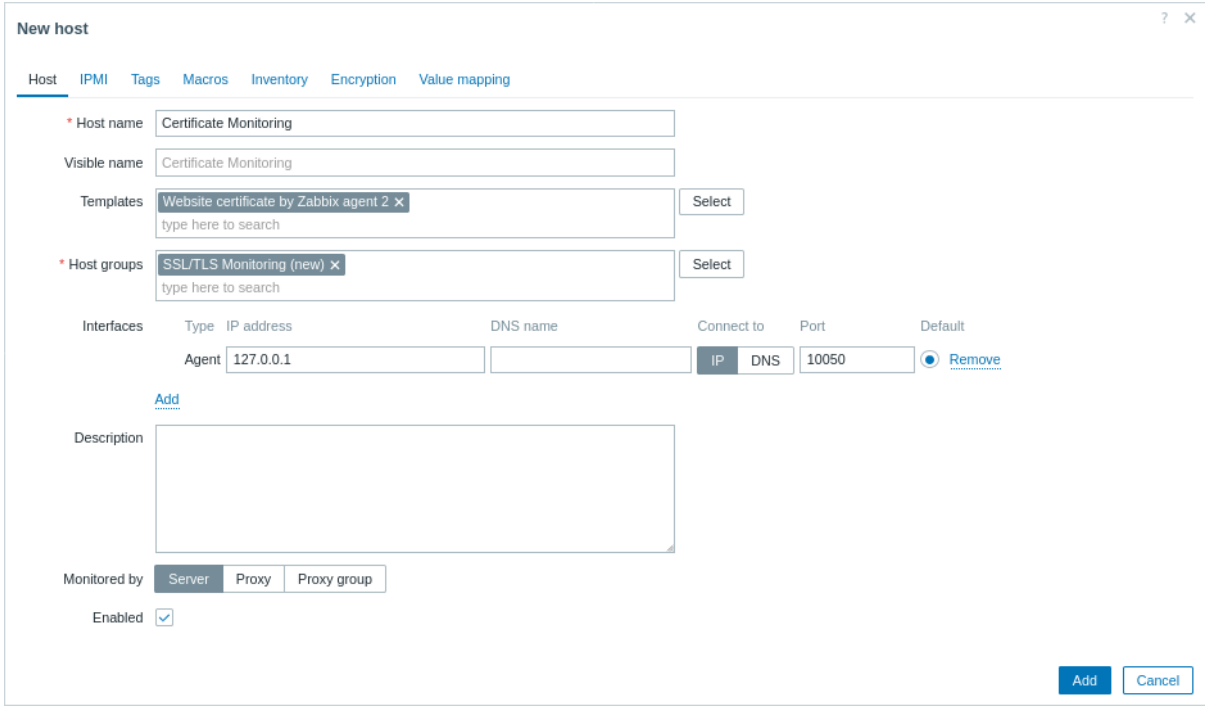
```
zabbix_get -s 127.0.0.1 -k web.certificate.get[<website_DNS_name>]
```

Agent Zabbix 2 domyślnie zawiera wtyczkę WebCertificate, dlatego nie jest wymagana osobna instalacja ani konfiguracja.

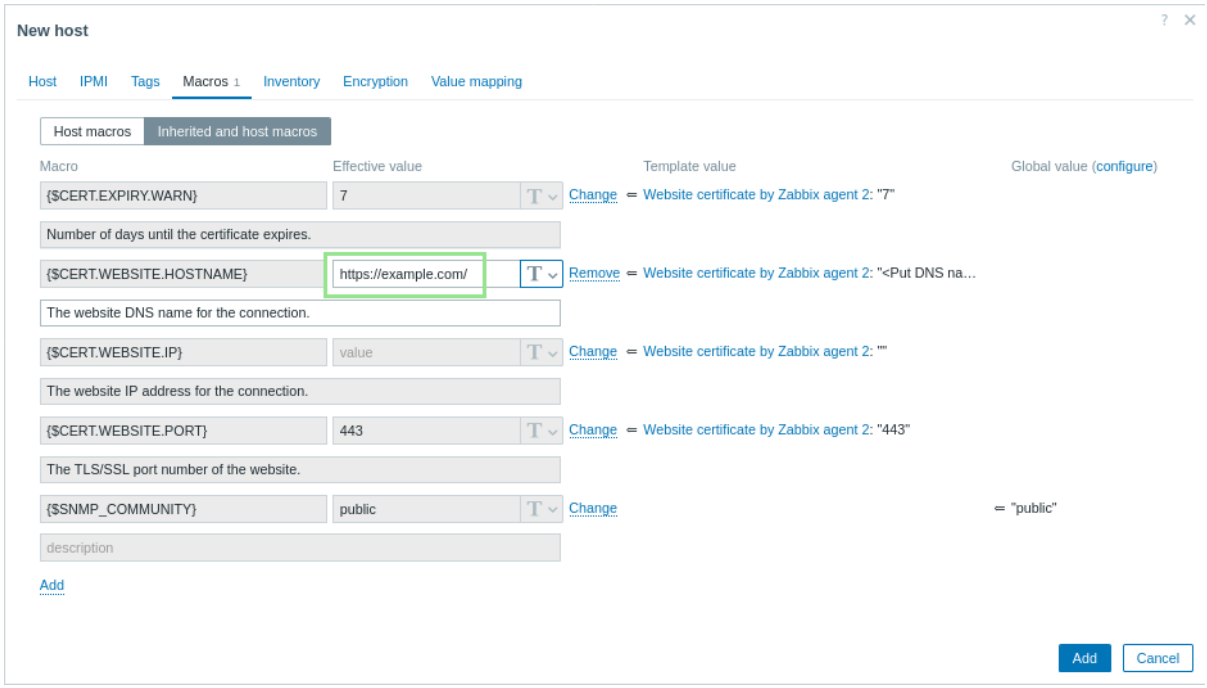
Konfiguracja frontend Zabbix

1. Zaloguj się do frontend Zabbix.
2. Przejdź do *Monitoring* > *Hosts*.
3. Kliknij istniejący host, dla którego chcesz monitorować certyfikaty witryn internetowych, lub w razie potrzeby utwórz host:

- W polu *Host name* wpisz nazwę hosta (np. ”Certificate Monitoring”).
- W polu *Templates* wpisz lub wybierz szablon ”Website certificate by Zabbix agent 2”, który zostanie powiązany z hostem.
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. ”SSL/TLS Monitoring”).
- W polu *Interfaces* dodaj interfejs typu ”Agent” i określ adres IP. W tym przykładzie użyto ”127.0.0.1”.



- Na karcie *Macros* przejdź do *Inherited and host macros*, znajdź następujące makra i kliknij *Change* obok wartości makra, aby ją zaktualizować:
  - `{$CERT.WEBSITE.HOSTNAME}` - wpisz jako wartość żądaną nazwę DNS witryny.



4. Kliknij *Add*, aby dodać host.

Aby monitorować wiele witryn internetowych, możesz podać w makrze `{$CERT.WEBSITE.HOSTNAME}` listę ich nazw hostów rozdzieloną przecinkami. Opcjonalnie możesz też podać w makrze `{$CERT.WEBSITE.PORT}` listę portów rozdzieloną przecinkami, gdzie każdy port odpowiada odpowiedniej nazwie hosta w tej samej kolejności. Na przykład, aby monitorować example.com i example.org na porcie 8443:

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

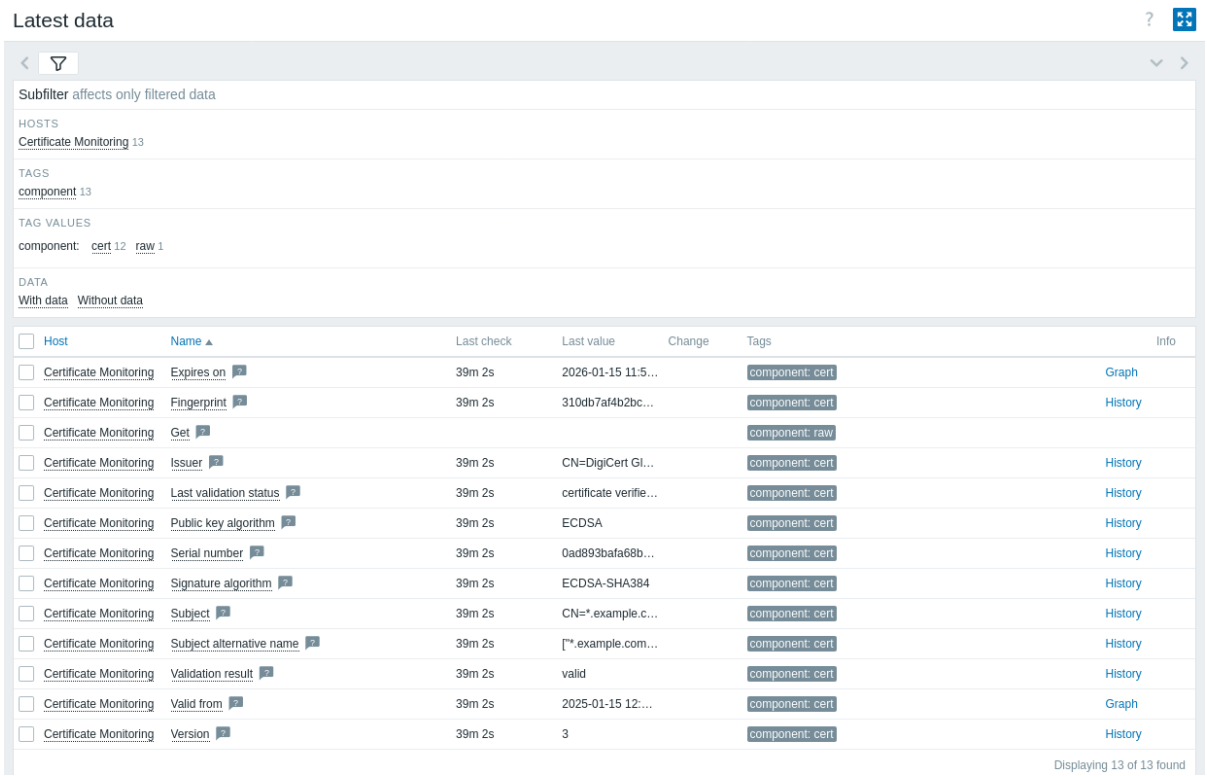
Dla każdej witryny określonej w makrze `{$CERT.WEBSITE.HOSTNAME}` Zabbix utworzy odpowiedni zestaw pozycji i wyzwalaczy. Umożliwia to indywidualne monitorowanie i generowanie alertów dla certyfikatu SSL każdej witryny.

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:…

Wyświetlanie zebranych metryk

Gratulacje! W tym momencie Zabbix już monitoruje wybrany certyfikat WWW.

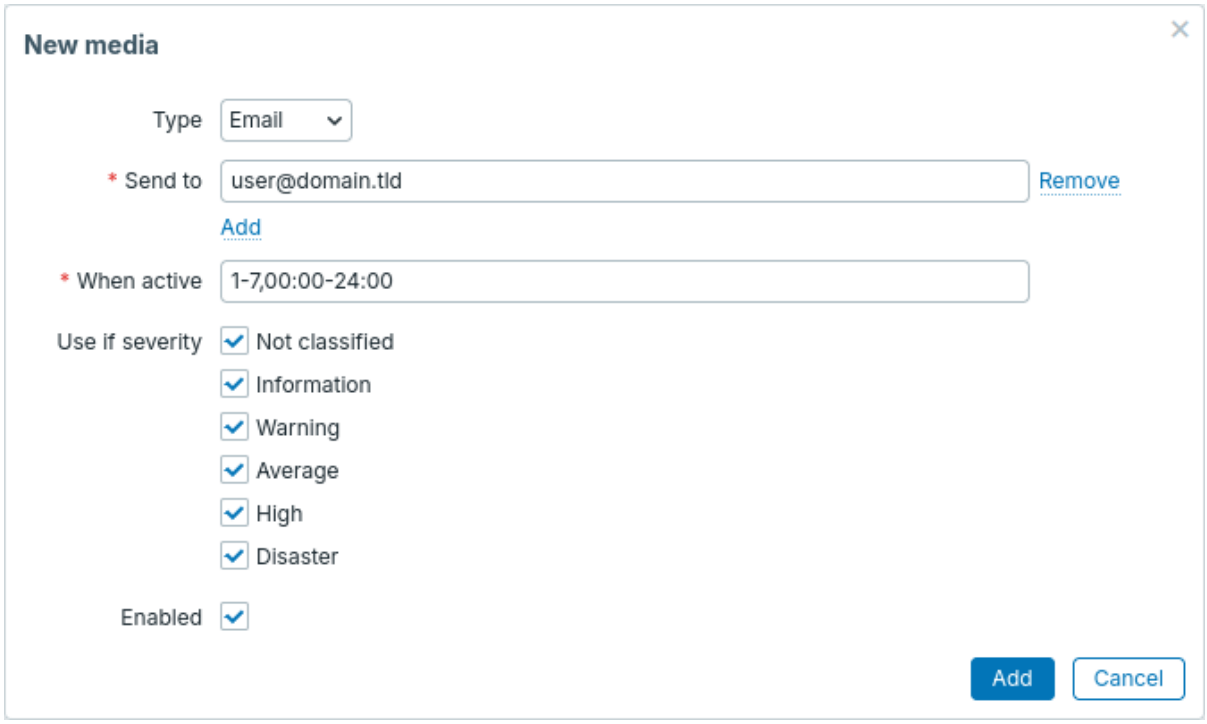
Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring > Hosts* i kliknij *Latest data* obok hosta, aby wyświetlić wszystkie najnowsze zebrane metryki na liście, takie jak data wygaśnięcia, wystawca i podmiot.



Konfiguracja alertów o problemach

Zabbix może powiadamiać o problemach z infrastrukturą. Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



2. Postępuj zgodnie z przewodnikiem Odbieranie powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Przetestuj swoją konfigurację

Aby przetestować swoją konfigurację, możemy zasymulować rzeczywisty problem, aktualizując konfigurację hosta w frontend Zabbix.

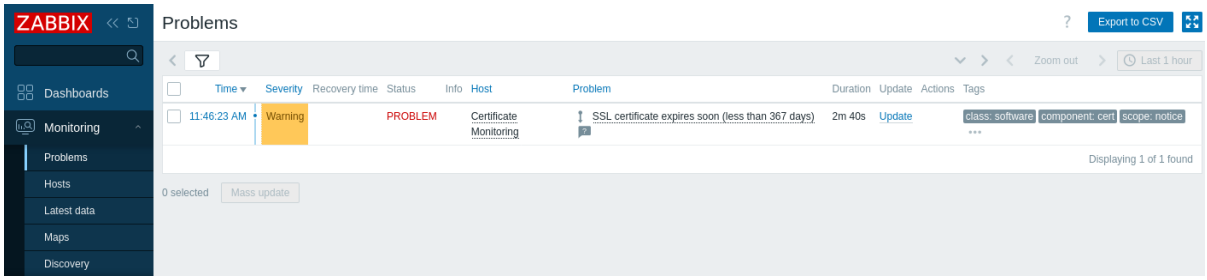
1. Otwórz konfigurację hosta ”Certificate Monitoring” w Zabbix.

2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.

3. Kliknij *Change* obok wcześniej skonfigurowanej wartości makra `{$CERT.EXPIRY.WARN}` i ustaw bardzo dużą liczbę dni (wartość większa niż 365 dni powinna być wystarczająca), aby otrzymać ostrzeżenie przed wygaśnięciem certyfikatu.

4. Kliknij *Update*, aby zaktualizować konfigurację hosta.

5. Po chwili Zabbix wykryje problem "SSL certificate expires soon" wraz z liczbą dni pozostałych do wygaśnięcia. Problem pojawi się w *Monitoring > Problems*.



Jeśli alerty są skonfigurowane, otrzymasz również powiadomienie o problemie.

6. Przywróć wartość makra do poprzedniej wartości, aby rozwiązać problem i kontynuować monitorowanie wartości certyfikatu.

Zobacz także

- Zabbix agent 2 - zawiera listę kluczy pozycji.
- Szablon *Website certificate by Zabbix agent 2* - dodatkowe informacje o szablonie *Website certificate by Zabbix agent 2*.
- Szablon *Website certificate by Zabbix agent 2 active* - dodatkowe informacje o szablonie *Website certificate by Zabbix agent 2 active*.
- Monitorowanie witryn internetowych za pomocą pozycji Browser - jak rozpocząć podstawowe monitorowanie witryn internetowych za pomocą pozycji Browser.

Monitoruj przełącznik sieciowy lub router za pomocą Zabbix

Wprowadzenie

Ten przewodnik przeprowadzi Cię przez kroki wymagane do rozpoczęcia podstawowego monitorowania przełącznika sieciowego lub routera za pomocą Zabbixa. Jako przykład użyto routera Cisco, ale procedura ma zastosowanie do dowolnego urządzenia sieciowego obsługującego SNMP.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbixa i administratorów sieci, którzy chcą szybko włączyć podstawowe monitorowanie urządzeń sieciowych. Jeśli potrzebujesz głębokiej personalizacji lub zaawansowanych opcji konfiguracji, zapoznaj się ze stroną SNMP agent lub sekcją Configuration podręcznika Zabbixa.

Wymagania wstępne

Przed kontynuowaniem upewnij się, że masz:

- Zabbix serwer i frontend Zabbixa zainstalowane: zainstaluj zgodnie z instrukcjami dla swojego systemu operacyjnego (zobacz Installation from packages oraz Web interface installation).
- Zabbix agent zainstalowany, jeśli monitorujesz lokalne metryki sieciowe.
- Urządzenie obsługujące SNMP: przełącznik sieciowy lub router (na przykład router Cisco) z włączonym SNMP.
- Zainstalowane pliki MIB: instalacja plików MIB umożliwia Zabbixowi tłumaczenie numerycznych OID-ów na czytelne dla człowieka nazwy i opisy. Bez odpowiedniego wsparcia MIB możesz widzieć tylko wartości numeryczne, co utrudnia konfigurację pozycji i rozwiązywanie problemów.

Aby zainstalować pliki MIB w Ubuntu:

1. Zainstaluj pakiet pobierający pliki MIB:

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

Jeśli musisz dodać MIB-y specyficzne dla producenta (np. Cisco, Juniper), umieść je w odpowiednim katalogu MIB:

- W systemach opartych na Linuksie typowe lokalizacje to `/usr/share/snmp/mibs/` lub `/usr/local/share/snmp/mibs/`.
- W instalacjach Zabbixa pliki MIB mogą być przechowywane w `/var/lib/zabbix/mibs/`.

Upewnij się, że zmienna środowiskowa `MIBDIRS` lub plik `snmp.conf` zawiera poprawną ścieżkę.

Aby sprawdzić, czy system rozpoznaje nowe MIB-y, użyj:

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

Szczegółowe instrukcje znajdziesz w dokumentacji biblioteki SNMP:

- Cisco MIBs
- Juniper MIBs

2. Edytuj `/etc/snmp/snmp.conf` i zakomentuj linię zaczynającą się od `mibs :`, aby umożliwić systemowi załadowanie wszystkich dostępnych MIB-ów.

3. Zweryfikuj działanie, uruchamiając `snmpwalk` (na przykład `snmpwalk -v 2c -c <your_community_string> <device_IP>`) i sprawdź, czy OID-y są wyświetlane z opisowymi nazwami.

W zależności od środowiska niektóre kroki w tym przewodniku mogą się nieznacznie różnić. Ten przewodnik opiera się na środowisku z systemem Ubuntu oraz monitorowanym urządzeniu sieciowym Cisco Catalyst 3750V2-24FS.

Zakłada się, że urządzenie sieciowe jest już fizycznie zainstalowane i podłączone.

Skonfiguruj urządzenie sieciowe (przykład routera Cisco)

Aby monitorować za pomocą SNMP, należy skonfigurować urządzenie sieciowe tak, aby zezwalało na zapytania SNMP. Poniższy przykład dotyczy SNMPv2 i nie uwzględnia istniejących ustawień. Uwaga: zastosowanie tych poleceń może zastąpić bieżące konfiguracje SNMP.

W przypadku routera Cisco konfiguracja zazwyczaj obejmuje kroki opisane poniżej.

Przykład SNMPv2

1. Włącz SNMP i ustaw ciąg community.

Zaloguj się do konsoli routera Cisco i przejdź do trybu konfiguracji:

```
configure terminal
```

Następnie włącz SNMP, określając tylko do odczytu ciąg community. Na przykład:

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

Zastąp `<your_community_string>` swoim bezpiecznym ciągiem community. Uwaga: opcja RO (Read-Only) umożliwia SNMP pobieranie danych z urządzenia, ale uniemożliwia wprowadzanie jakichkolwiek zmian konfiguracji.

Ze względów bezpieczeństwa zaleca się ograniczenie dostępu SNMP wyłącznie do niezbędnych urządzeń. Więcej informacji na temat konfigurowania list kontroli dostępu (ACL) znajdziesz w oficjalnej dokumentacji Cisco.

2. Zapisz konfigurację.

Zapisz zmiany, aby ustawienia SNMP pozostały zachowane po ponownym uruchomieniu:

```
write memory
```

Przykład SNMPv3

SNMPv3 zapewnia zwiększone bezpieczeństwo dzięki uwierzytelnianiu i szyfrowaniu. Jego konfiguracja jest bezpieczniejsza niż SNMPv2 i powinna zostać zweryfikowana na podstawie dokumentacji właściwej dla danego urządzenia.

1. Utwórz grupę SNMP.

Skonfiguruj grupę SNMPv3 z włączoną prywatnością (szyfrowaniem):

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

2. Utwórz użytkownika SNMP.

Dodaj użytkownika SNMPv3 z uwierzytelnianiem i prywatnością. Zastąp symbole zastępcze wartościami, których chcesz użyć:

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

3. Zapisz konfigurację:

```
write memory
```

Więcej informacji lub instrukcje specyficzne dla danego modelu można znaleźć w zewnętrznych samouczkach konfiguracji SNMP firmy Cisco. Niniejszy przewodnik zawiera jednak podstawowe kroki umożliwiające włączenie monitorowania SNMP.

Konfiguracja frontend Zabbixa

Utwórz host w frontend Zabbix

1. Zaloguj się do frontend Zabbix.

2. Dodaj nowy host.

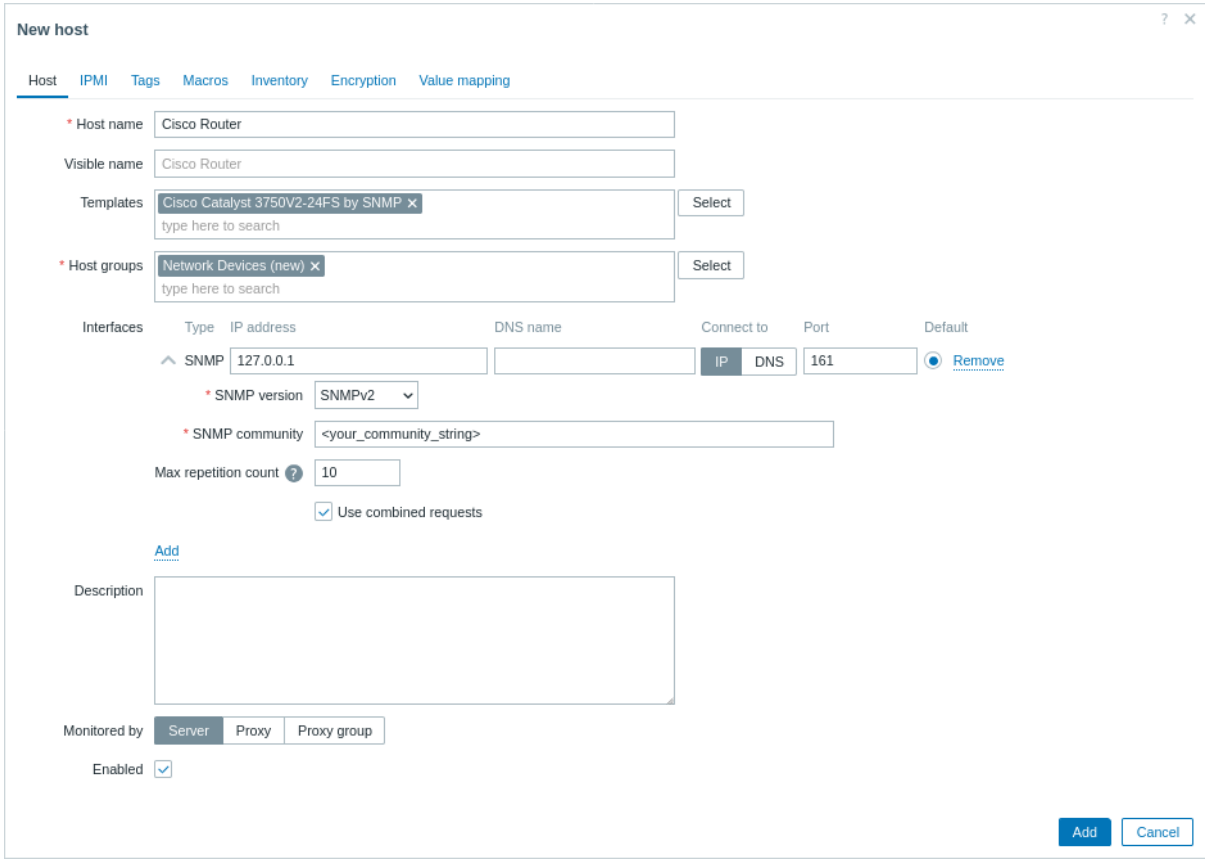
Przejdź do *Data collection* > *Hosts* i kliknij *Create host*.

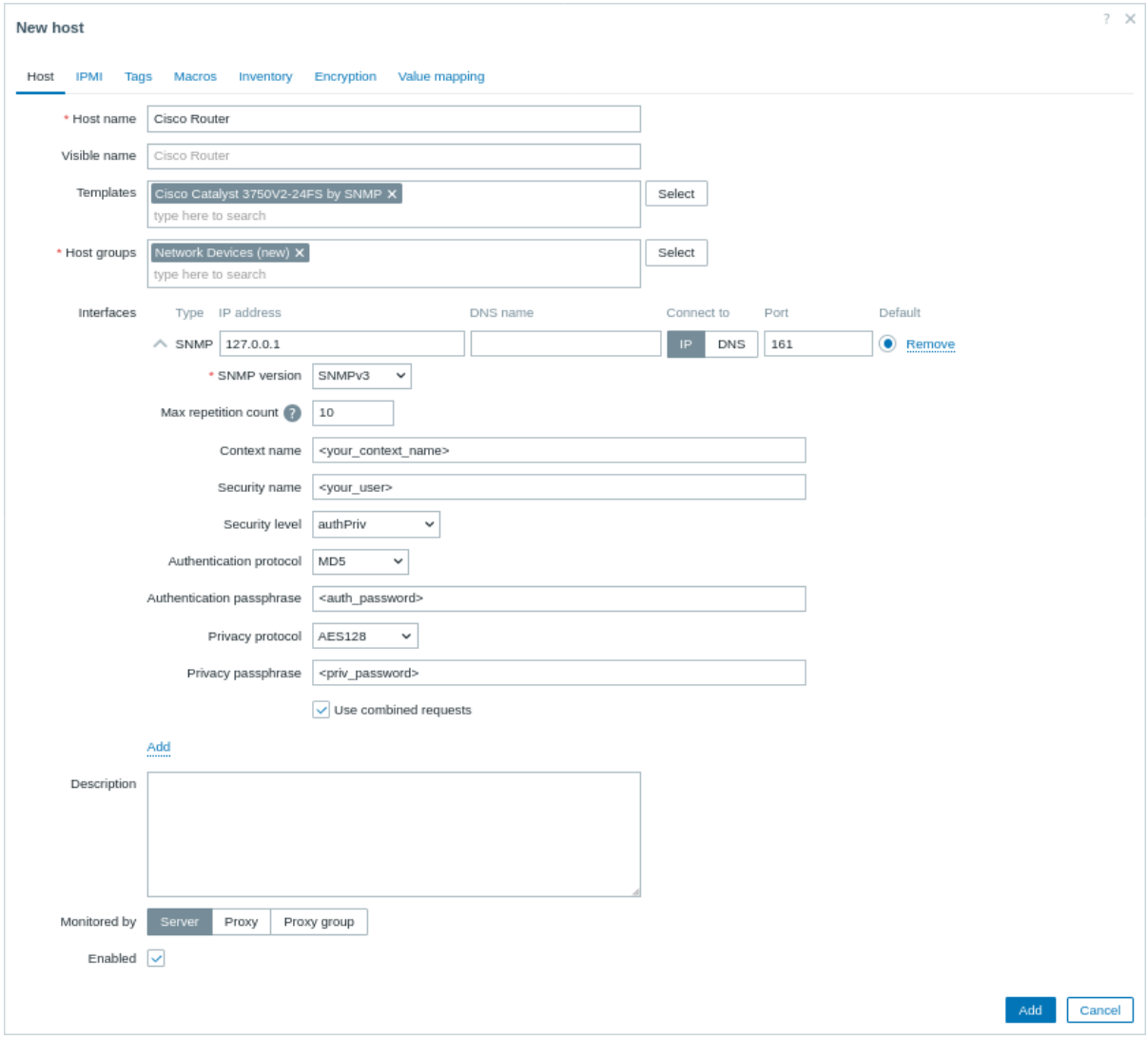
- *Host name*: wpisz nazwę swojego urządzenia (np. ”Cisco Router”).
- Host groups: wybierz istniejącą grupę lub utwórz nową, na przykład ”Network Devices”.
- Interfaces:
  - Kliknij *Add* w sekcji Interfaces.
  - Jako typ interfejsu wybierz *SNMP*.
  - Wprowadź adres IP lub nazwę DNS swojego routera Cisco.
  - Ustaw domyślny port SNMP (zwykle 161).
  - Użyj menu rozwijanego, aby wybrać odpowiednią wersję SNMP (np. SNMPv2).
  - Dla SNMPv1/v2 wpisz ciąg community w polu *SNMP community*. Dla SNMPv3 zostaną wyświetlone dodatkowe dane uwierzytelniające (*Context name*, *Security name* i *Security level* itp.).

3. Połącz szablony

W polu *Templates* wybierz szablon SNMP, który najlepiej odpowiada Twojemu urządzeniu. Zabbix udostępnia zestaw gotowych szablonów SNMP dla wielu rodzin urządzeń. Na przykład, jeśli monitorujesz urządzenie Cisco, wybierz szablon odpowiadający systemowi operacyjnemu lub modelowi urządzenia (na przykład Cisco IOS SNMP lub Cisco Catalyst 3750<device model> SNMP).

4. Kliknij *Add*, aby zapisać host.



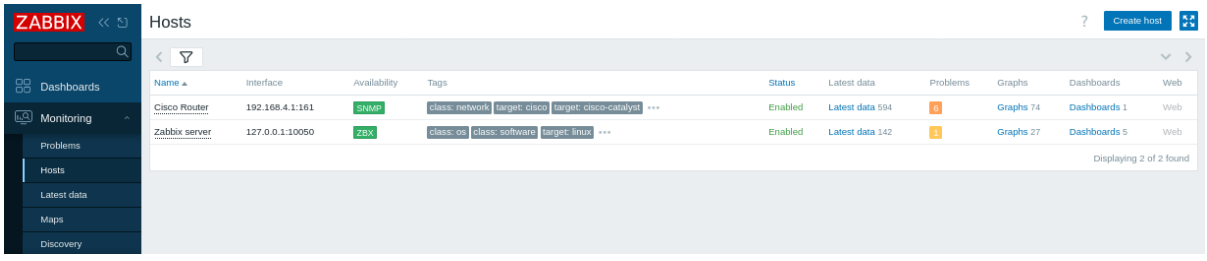


Wyświetlanie zebranych metryk

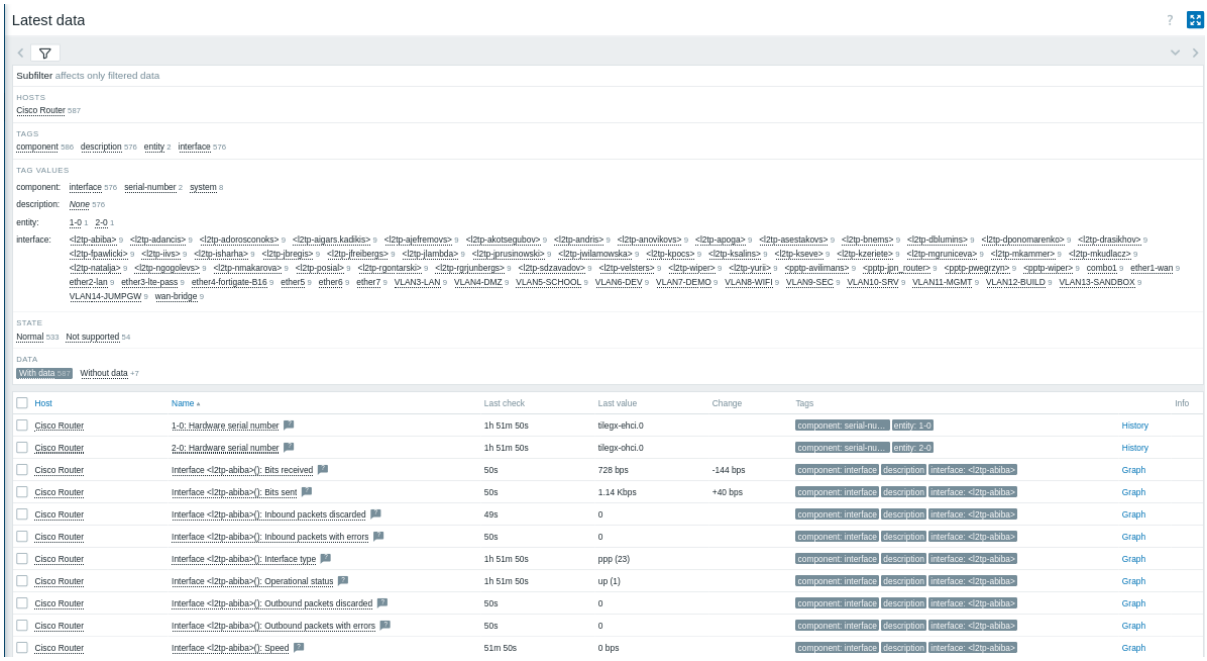
Gratulacje! Zabbix jest teraz skonfigurowany do monitorowania Twojego urządzenia sieciowego.

Najnowsze dane:

- Przejdź do Monitoring > Latest data w frontendzie Zabbixa.



- Wybierz host ”Cisco Router” (lub wykryte hosty), aby wyświetlić metryki, takie jak czas działania sprzętu i sieci, utrata ICMP, ping, czas odpowiedzi itp.



- Wykresy i ekrany:

Aby wizualizować dane wydajności, kliknij *Graphs* obok pozycji SNMP, aby zobaczyć szczegółowe metryki.

Następnie możesz:

- Dodać niestandardowe pozycje SNMP, aby monitorować dodatkowe metryki.
- Skonfigurować alerty o problemach, aby otrzymywać powiadomienia o potencjalnych problemach.

Tworzenie pozycji SNMP

Gdy host jest już skonfigurowany, możesz utworzyć pozycje do monitorowania określonych metryk. Uwaga: ten krok jest opcjonalny, jeśli używasz szablonu, ponieważ szablony zawierają już domyślne zestawy pozycji.

1. Zidentyfikuj OID SNMP:

Użyj polecenia `snmpwalk`, aby wyświetlić dostępne OID-y na urządzeniu. Na przykład:

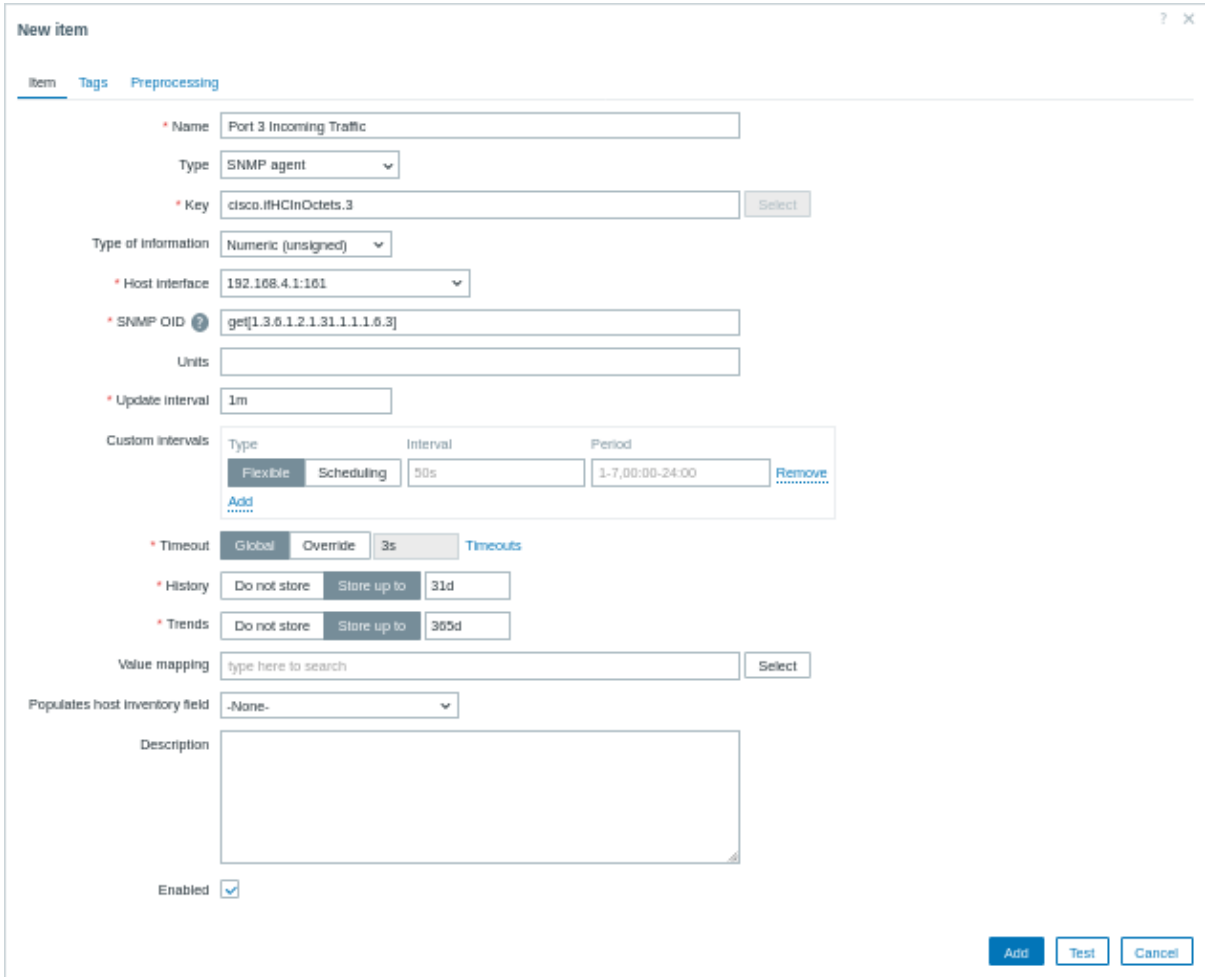
```
snmpwalk -v 2c -c <your_community_string> <device_IP> .
```

Znajdź OID dla metryki, którą chcesz monitorować (na przykład IF-MIB::ifHCInOctets.3 dla ruchu przychodzącego na porcie 3). Aby uzyskać numeryczny OID, możesz użyć:

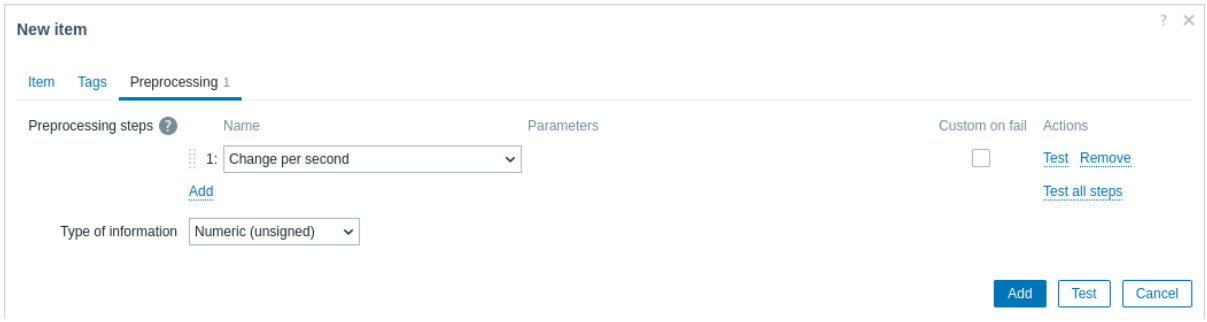
```
snmpget -v 2c -c <your_community_string> -On <device_IP> IF-MIB::ifHCInOctets.3
```

2. Utwórz pozycję SNMP:

- Przejdź do *Zbieranie danych* > *Hosts* i kliknij kartę *Pozycje* dla hosta SNMP, a następnie kliknij *Utwórz pozycję*.
- *Nazwa*: wprowadź opisową nazwę (np. ”Ruch przychodzący na porcie 3”).
- *Typ*: wybierz *SNMP agent*.
- *Klucz*: podaj czytelny klucz (np. `cisco.ifHCInOctets.3`).
- *Interfejs hosta*: upewnij się, że wybrano interfejs SNMP.
- *OID SNMP*: wprowadź OID, używając jednego z obsługiwanych formatów, na przykład:
  - `get[1.3.6.1.2.1.31.1.1.1.6.3]` dla pojedynczej wartości;
  - `walk[1.3.6.1.2.1.31.1.1.1.6.3]` aby asynchronicznie pobrać poddrzewo wartości.



- *Przetwarzanie wstępne* (jeśli potrzebne): jeśli pozycja zwraca licznik narastający (na przykład ruch interfejsu), przejdź do karty *Przetwarzanie wstępne* i dodaj krok przetwarzania, taki jak "Zmiana na sekundę", aby obliczyć tempo.



Aby pobrać wiele wartości w ramach jednej transakcji SNMP, możesz określić kilka OID-ów, używając składni `walk[OID1,OID2,...]`.

Tłumaczenie OID-ów między nazwami numerycznymi a nazwami MIB

Podczas pracy z SNMP może być konieczne konwertowanie między numerycznymi OID-ami a odpowiadającymi im nazwami MIB. Takie tłumaczenie ułatwia identyfikowanie metryk i rozwiązywanie problemów.

- Tłumaczenie nazwy MIB na numeryczny OID: użyj polecenia `snmptranslate` z opcją `-On`. Na przykład, aby przetłumaczyć nazwę MIB `IF-MIB::ifHCInOctets.3` na jej numeryczny OID, uruchom:

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

To polecenie może zwrócić:

```
.1.3.6.1.2.1.31.1.1.1.6.3
```

- Tłumaczenie numerycznego OID na jego nazwę MIB: użyj polecenia `snmptranslate` z opcją `-IR` (lub `-m ALL`), aby odwrócić tłumaczenie. Na przykład, aby przetłumaczyć numeryczny OID `.1.3.6.1.2.1.31.1.1.1.6.3` z powrotem na jego nazwę MIB, uruchom:

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

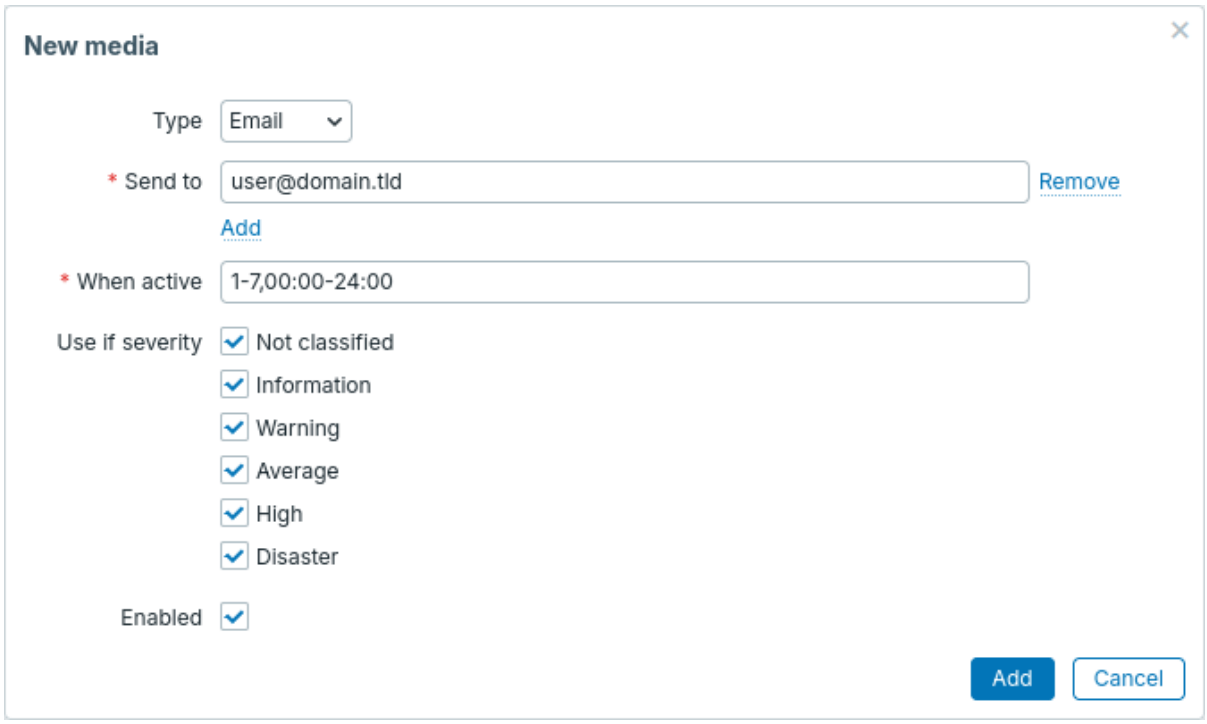
To polecenie może zwrócić:

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

Konfiguracja alertów o problemach

Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do *Ustawienia użytkownika > Profil*, przejdź do zakładki *Media* i dodaj swój adres e-mail.



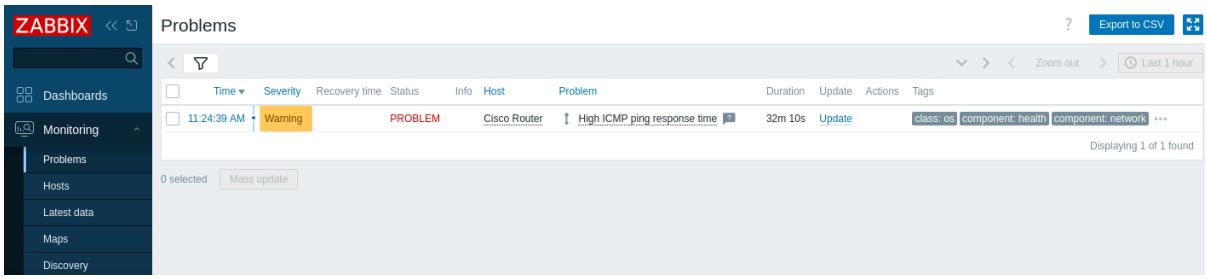
2. Postępuj zgodnie z przewodnikiem dotyczącym otrzymywania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mailem.

Przetestuj swoją konfigurację

Aby upewnić się, że Zabbix prawidłowo wykrywa problemy z wydajnością sieci, zasymuluj rzeczywisty problem, zwiększając próg czasu odpowiedzi ping ICMP.

1. Otwórz konfigurację hosta ”Cisco Router” w Zabbix.
2. Przejdź do zakładki *Macros* i wybierz *Inherited and host macros*.
3. Znajdź makro `{$ICMP_RESPONSE_TIME_WARN}` (lub podobne makro progu czasu odpowiedzi).
4. Ustaw bardzo niską wartość (np. 0.001), aby wyzwolić alert, gdy czas odpowiedzi ping przekroczy tę wartość.
5. Kliknij *Update*, aby zastosować zmiany.
6. Poczekaj kilka chwil, aż Zabbix wykryje zasymulowany problem.
7. Przejdź do *Monitoring > Problems*, aby sprawdzić, czy pojawił się alert (np. ”High ICMP ping response time”).



Jeśli alerty są skonfigurowane, powinieneś również otrzymać alert o problemie.

8. Przywróć pierwotną wartość makra i kliknij *Update*, aby zapisać zmiany.
9. Potwierdź, że problem został rozwiązany i zniknął z sekcji *Problems*.

Rozwiązywanie problemów z monitorowaniem SNMP

Jeśli zauważysz, że ikona SNMP w frontend Zabbixa jest CZERWONA lub nie są zbierane żadne dane, spróbuj wykonać następujące kroki:

1. Sprawdź łączność SNMP.

Dla SNMPv2 uruchom następujące polecenie z serwera Zabbixa:

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

To polecenie sprawdza, czy urządzenie odpowiada na zapytania SNMP.

Dla SNMPv3 dołącz odpowiednie poświadczenia SNMPv3:

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

To potwierdza, że poświadczenia SNMPv3 są poprawne, a urządzenie odpowiada w bezpieczny sposób.

2. Upewnij się, że pliki MIB są zainstalowane i włączone zgodnie z opisem w wymaganiach wstępnych. Aby to sprawdzić, następujące polecenie nie powinno zwracać błędu podczas odpytywania urządzenia sieciowego:

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

Powinno to zwrócić przetłumaczone OID bez błędów.

3. Potwierdź, że wersja SNMP i poświadczenia skonfigurowane w Zabbiksie są zgodne z tymi ustawionymi na urządzeniu. Na przykład przejrzyj ustawienia SNMP w konfiguracji hosta Zabbixa i porównaj je z konfiguracją urządzenia. Na urządzeniu Cisco możesz sprawdzić ustawienia SNMP, uruchamiając:

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

Zapewnia to, że community string (dla SNMPv2) lub dane użytkownika SNMPv3 są poprawne.

4. Sprawdź, czy SNMP jest poprawnie włączone na urządzeniu sieciowym. Na routerze Cisco zaloguj się do konsoli i uruchom:

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

To polecenie wyświetla aktywną konfigurację SNMP i pomaga potwierdzić, że SNMP jest poprawnie skonfigurowane.

5. Upewnij się, że żadne zapory sieciowe ani problemy sieciowe nie blokują ruchu SNMP (zwykle na porcie 161) między serwerem Zabbixa a urządzeniem. Możesz przetestować łączność za pomocą:

nc -zv <device_IP> 161

`nc -zv` sprawdza, czy port 161 jest otwarty i nasłuchuje na urządzeniu.

Dodatkowo, jeśli używasz UFW na Ubuntu, sprawdź stan zapory:

```
sudo ufw status
```

Albo, w przypadku iptables:

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

6. Przejrzyj pliki dziennika serwera Zabbixa pod kątem błędów związanych z SNMP, aby pomóc zlokalizować problem:

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

`tail -f` umożliwia monitorowanie aktualizacji dziennika w czasie rzeczywistym.

Zobacz także:

- Tworzenie pozycji - dowiedz się, jak dodawać dodatkowe metryki.
- SNMP agent - dodatkowe informacje o monitorowaniu SNMP w Zabbix.
- Ustandaryzowane szablony dla urządzeń sieciowych - informacje o dostępnych szablonach SNMP.
- Odnajdywanie SNMP OID-ów - dodatkowe informacje o wykrywaniu SNMP na przełączniku.
- Konfigurowanie reguły wykrywania sieci - dodatkowe informacje o tym, jak skonfigurować regułę wykrywania sieci używaną przez Zabbix do wykrywania hostów i usług.

Monitorowanie dziennika zdarzeń systemu Windows za pomocą aktywnych kontroli

Wprowadzenie

Ten przewodnik wyjaśnia, jak monitorować dzienniki zdarzeń systemu Windows za pomocą Zabbixa, korzystając z aktywnych kontroli. Dzięki kluczom pozycji specyficznym dla systemu Windows w Zabbixie możesz zbierać i analizować krytyczne zdarzenia (takie jak nieudane próby logowania, błędy systemowe itp.) w czasie rzeczywistym.

Dla kogo jest ten przewodnik

Ten przewodnik jest przeznaczony dla nowych użytkowników Zabbixa oraz administratorów sieci, którzy chcą monitorować dzienniki zdarzeń systemu Windows. Informacje o zaawansowanych opcjach konfiguracji znajdziesz w dokumentacji dotyczącej kluczy pozycji specyficznych dla systemu Windows.

Wymagania wstępne

Przed rozpoczęciem korzystania z tego przewodnika musisz pobrać i zainstalować serwer Zabbix oraz frontend Zabbixa zgodnie z instrukcjami dla swojego systemu operacyjnego. Musisz również pobrać i zainstalować Zabbix agent pobrany i zainstalowany na komputerze z systemem Windows, który chcesz monitorować.

Konfigurowanie agenta Zabbix do monitorowania dziennika zdarzeń systemu Windows

1. Otwórz plik konfiguracyjny agenta na swoim hoście Windows i upewnij się, że parametr `ServerActive` jest ustawiony na adres IP serwera Zabbix, a parametr `Hostname` odpowiada nazwie hosta, która zostanie zdefiniowana w frontend Zabbix. Umożliwia to agentowi żądanie aktywnych kontroli dla swojego hosta i z określonego serwera Zabbix. Na przykład:

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

2. Uruchom ponownie usługę agenta Zabbix, aby zastosować zmiany:

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

3. Sprawdź, czy host Windows działa:

- Upewnij się, że usługa agenta Zabbix działa na hoście Windows.
- Sprawdź, czy host Windows może połączyć się z serwerem Zabbix na porcie 10051. Aby przetestować łączność z hosta Windows, otwórz PowerShell i uruchom następujące polecenie:

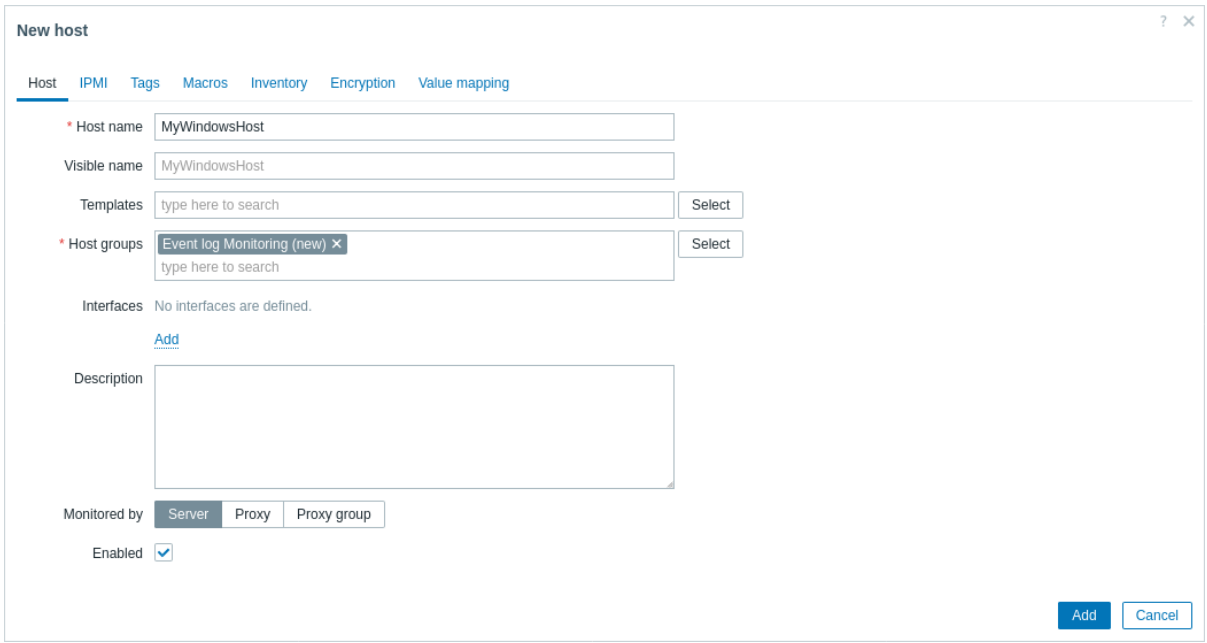
```
Test-NetConnection -ComputerName <Zabbix-server-IP> -Port 10051
```

Skonfiguruj frontend Zabbix

1. Przejdź do *Data collection* > *Hosts* i utwórz host:

- W polu *Host name* wpisz nazwę hosta (np. ”MyWindowsHost”).
- W polu *Host groups* wpisz lub wybierz grupę hostów (np. ”Event log Monitoring”).
- Kliknij *Add*, aby zapisać skonfigurowany host.

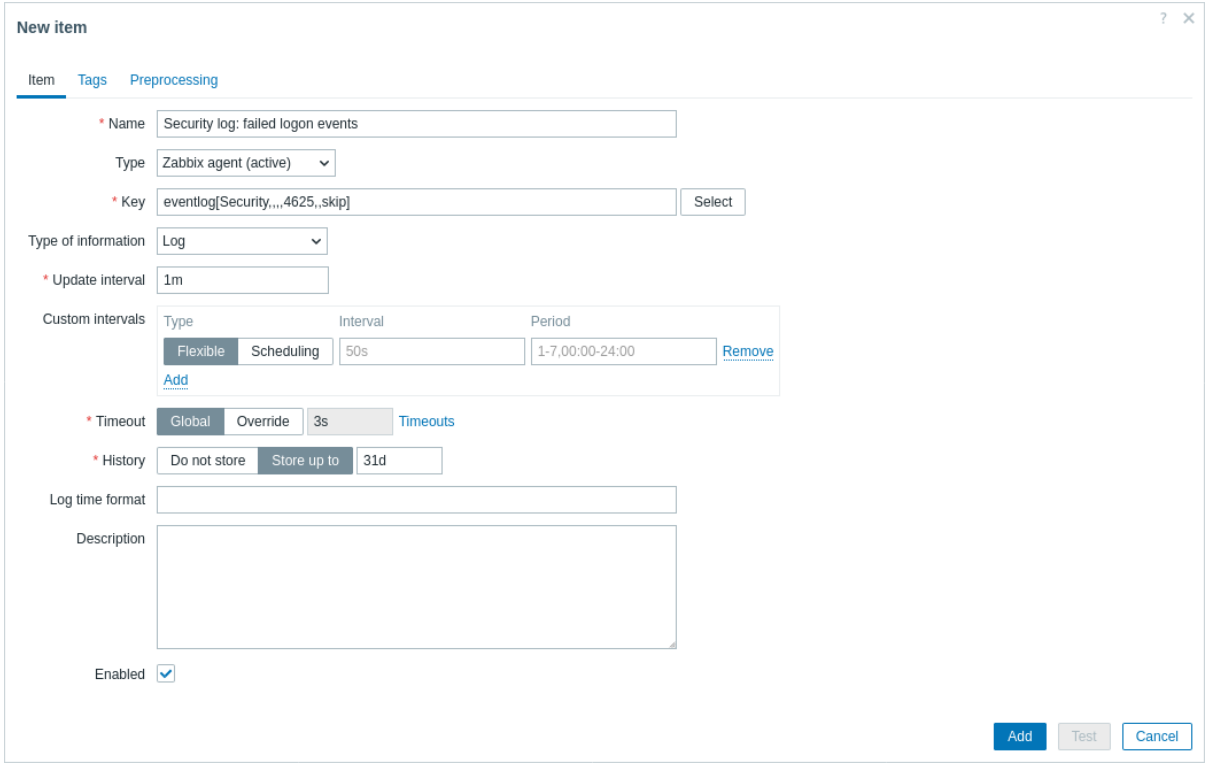
Note:
W polu *Templates* możesz dodać szablon ”Windows by Zabbix agent active”, aby ułatwić rozwiązywanie problemów poprzez obserwację, czy inne aktywne pozycje na tym samym hoście są aktualizowane.



2. Utwórz nową pozycję z następującymi parametrami:

- W polu *Name* wpisz opisową nazwę pozycji (np. ”Security log: failed logon events”).
- W rozwijanym polu *Type* wybierz ”Zabbix agent (active)” (wymagane do monitorowania dziennika zdarzeń).

- W polu *Key* użyj klucza pozycji eventlog. Na przykład, aby monitorować nieudane próby logowania (Event ID: 4625) w dzienniku Security i ignorować wpisy starsze niż ostatnie sprawdzenie pozycji (używając parametru `skip`), wpisz następujący klucz pozycji: `eventlog[Security,,,,4625,,skip]`
- W rozwijanym polu *Type of information* wybierz "Log".



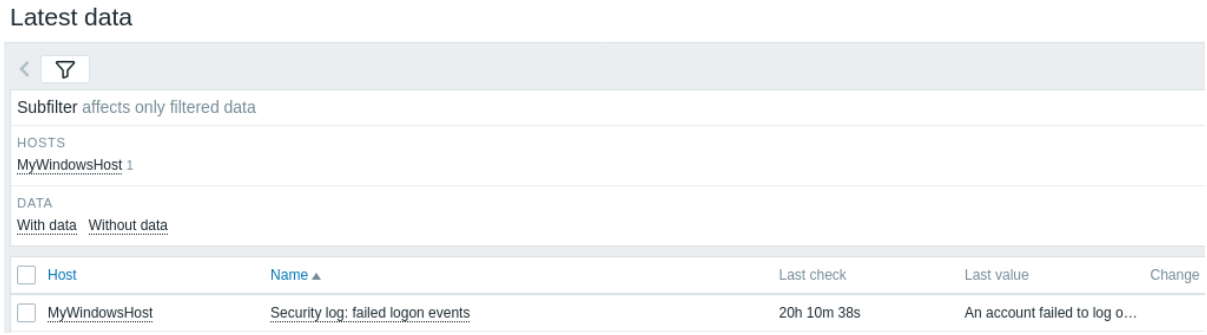
3. Kliknij *Add*, aby zapisać pozycję.

Testowanie i przeglądanie zebranych metryk

Gratulacje! Zabbix jest teraz skonfigurowany do zbierania dzienników zdarzeń systemu Windows. Aby sprawdzić, czy dzienniki zdarzeń są zbierane, możesz przetestować pozycję "Security log: failed logon events", wylogowując się ze swojego konta Windows i próbując zalogować się przy użyciu nieprawidłowych danych uwierzytelniających.

Następnie wyświetl zebrane dzienniki w frontend Zabbix:

1. Przejdź do *Monitoring* > *Latest data* w frontend Zabbix.



2. Przefiltruj według hosta "MyWindowsHost" w polu *Name*.

3. Kliknij *History*, aby wyświetlić zarejestrowane wartości dziennika.

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. Jeśli wartości dziennika są nieobecne, przejdź do sekcji Troubleshooting w tym przewodniku.

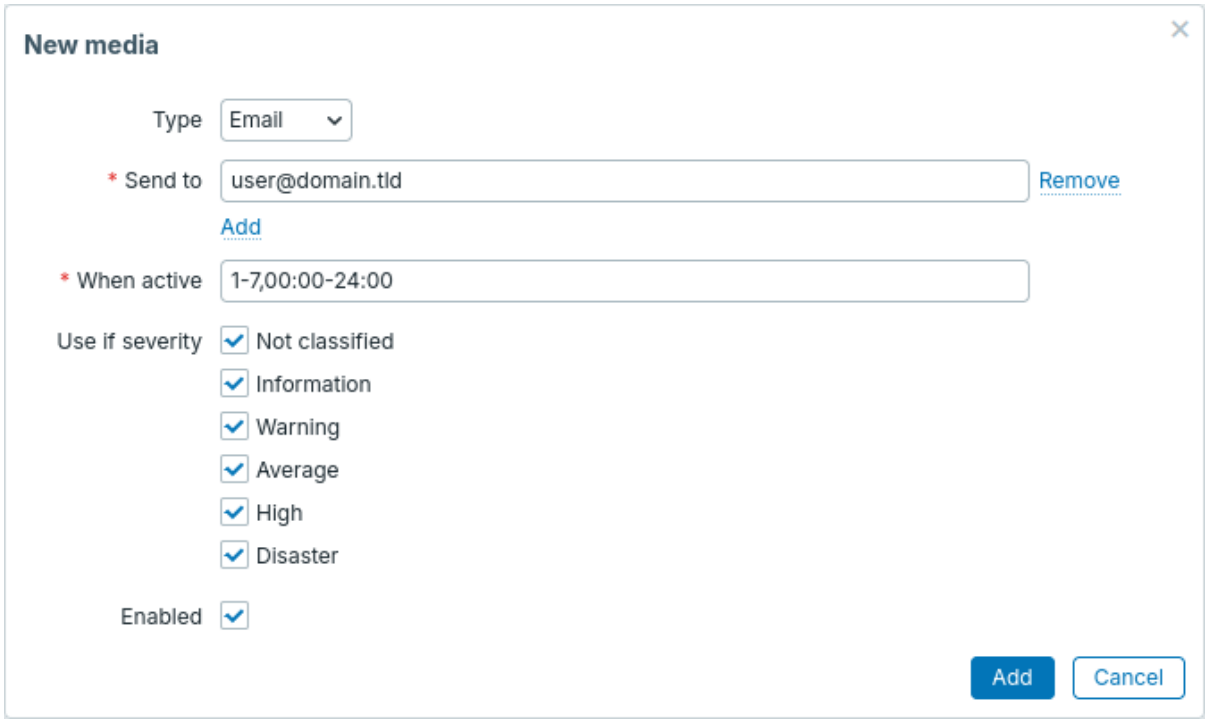
Konfiguracja alertów o problemach

Ten przewodnik zawiera podstawowe kroki konfiguracji wysyłania alertów e-mail.

1. Przejdź do Data collection > Hosts, aby zdefiniować wyzwalacz, który uruchomi się, gdy pozycja dziennika zdarzeń zapisze interesujący Cię wzorzec. Na przykład, aby wykrywać nieudane próby logowania w dzienniku Security, użyj funkcji find():

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

2. Przejdź do *User settings > Profile*, przełącz się na kartę *Media* i dodaj swój adres e-mail.



3. Postępuj zgodnie z przewodnikiem dotyczącym otrzymywania powiadomienia o problemie.

Następnym razem, gdy Zabbix wykryje problem, powinieneś otrzymać alert e-mail.

Rozwiązywanie problemów

Jeśli napotkasz problemy z zbieraniem lub wyświetlaniem dzienników zdarzeń systemu Windows, skorzystaj z poniższych wskazówek, aby zidentyfikować i rozwiązać typowe problemy:

1. Na serwerze Zabbix (Linux) wyświetl reguły iptables za pomocą następującego polecenia:

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

i sprawdź, czy istnieje reguła ACCEPT dla portu TCP 10051.

2. Upewnij się, że klucz `eventlog[...]` używa dokładnej nazwy dziennika (z uwzględnieniem wielkości liter), identyfikatora zdarzenia, trybu (np. skip) oraz innych parametrów dokładnie tak, jak pokazano w Windows-specific item keys.

Zobacz także:

- Tworzenie pozycji - dowiedz się, jak dodawać dodatkowe metryki.
- Agent Zabbix w Microsoft Windows - szczegółowe instrukcje instalacji.
- Monitorowanie Windows za pomocą agenta Zabbix - kompleksowy przewodnik po konfiguracji podstawowego monitorowania komputerów z systemem Windows przy użyciu agenta Zabbix.
- Klucze pozycji specyficzne dla Windows - szczegółowe informacje o kluczach pozycji specyficznych dla Windows obsługiwanych przez agenty Zabbix, w tym dotyczących monitorowania dziennika zdarzeń.
- Monitorowanie plików dziennika - instrukcje dotyczące konfiguracji Zabbix do scentralizowanego monitorowania i analizy plików dziennika, mające zastosowanie do dzienników zdarzeń Windows.

Instalacja: dodatkowe przewodniki

Jeśli szukasz najłatwiejszego sposobu instalacji Zabbix, zapoznaj się z sekcją Instalacja i pierwsze kroki.

Poniższe przewodniki opisują **dodatkowe** lub specyficzne dla systemu sposoby instalacji Zabbix albo jego komponentów:

- Instalacja z kontenerów
- Instalacja ze źródeł
- Uwagi dotyczące instalacji z pakietów
- Instalacja agent Zabbix z użyciem Windows MSI
- Kompilowanie agent Zabbix 2 w systemie Windows
- Kompilowanie agent Zabbix w systemie Windows
- Instalacja agent w systemie macOS z pakietu PKG
- Agent Zabbix w systemie IBM AIX
- Kompilowanie agent Zabbix w systemie macOS
- Urządzenie Zabbix
- Przewodnik wdrażania serwera Zabbix w AWS
- Przewodnik wdrażania proxy Zabbix w AWS
- Przewodnik wdrażania serwera Zabbix w Azure
- Przewodnik wdrażania proxy Zabbix w Azure
- Przewodnik wdrażania w Google Cloud

Instalacja z kontenerów

Przegląd

Ta strona opisuje, jak wdrożyć Zabbix za pomocą Docker lub Docker Compose.

Docker Compose to najszybszy sposób wdrożenia Zabbix. Odczytuje plik konfiguracyjny i automatycznie uruchamia wszystkie kontenery składające się na kompletne środowisko Zabbix, we właściwej kolejności.

Docker (ręczne wdrożenie) osiąga ten sam rezultat, wdrażając każdy komponent krok po kroku.

Wymagania wstępne

Przed rozpoczęciem upewnij się, że w systemie jest zainstalowany Docker (1.12.0 lub nowszy). Jeśli nie jest zainstalowany, postępuj zgodnie z instrukcją instalacji platformy Docker. W przypadku wdrażania za pomocą Docker Compose wymagany jest również Docker Compose (2.24.0 lub nowszy).

Niektóre komponenty Zabbix wymagają otwarcia określonych portów na hoście, na którym działa Docker (np. 10051/TCP dla serwera Zabbix, 162/UDP dla pułapek SNMP). Pełną listę portów używanych przez komponenty Zabbix można znaleźć w sekcji Wymagania. W przypadku serwera Zabbix i agenta domyślny port można zmienić, ustawiając zmienną środowiskową `ZBX_LISTENPORT` w odpowiednim kontenerze.

Dostępne obrazy Docker

Zabbix udostępnia obraz Docker dla każdego komponentu Zabbix. Wszystkie obrazy są publikowane w serwisie Docker Hub. Każdy obraz służy do utworzenia kontenera uruchamiającego dany komponent.

Komponent Zabbix	Obraz Docker
Agent	zabbix/zabbix-agent
Agent 2	zabbix/zabbix-agent2
Serwer (MySQL)	zabbix/zabbix-server-mysql
Serwer (PostgreSQL)	zabbix/zabbix-server-pgsql
Frontend (Apache + MySQL)	zabbix/zabbix-web-apache-mysql
Frontend (Apache + PostgreSQL)	zabbix/zabbix-web-apache-pgsql
Frontend (Nginx + MySQL)	zabbix/zabbix-web-nginx-mysql
Frontend (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
Usługa web	zabbix/zabbix-web-service
Pułapki SNMP	zabbix/zabbix-snmptraps
Serwer MCP	zabbix/zabbix-mcp-server

Note:
Aby używać pułapek SNMP, kontener pułapek SNMP musi współdzielić wolumin z kontenerem serwera Zabbix lub proxy (zobacz jak używać tego obrazu w serwisie Docker Hub oraz poniższy przykład).

Wszystkie obrazy komponentów Zabbix w serwisie Docker Hub są oparte na najnowszych głównych wersjach obsługiwanych systemów operacyjnych. Obrazy te są automatycznie przebudowywane po zaktualizowaniu bazowych obrazów systemu operacyjnego.

Tagi obrazów

Każdy obraz obsługuje tagi umożliwiające wybór bazowego systemu operacyjnego i wersji Zabbix:

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

Obsługiwane wartości `<os>`:

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

Obsługiwane wartości `<version>`:

- `latest` - Najnowsza stabilna wersja Zabbix na Alpine Linux
- `<os>-latest` - Najnowsza stabilna wersja Zabbix w wybranym systemie operacyjnym
- `<os>-trunk` - Najnowsza wersja rozwojowa (nightly) w wybranym systemie operacyjnym
- `<os>-X.X-latest` - Najnowsze wydanie poboczne Zabbix dla określonej wersji głównej Zabbix w wybranym systemie operacyjnym
- `<os>-X.X.*` - Określone wydanie poboczne Zabbix w wybranym systemie operacyjnym

Przykłady:

```
#### Najnowszy stabilny Zabbix proxy (SQLite3) na Alpine Linux:
docker pull zabbix/zabbix-proxy-sqlite3:latest

#### Najnowszy stabilny Zabbix proxy (SQLite3) na Ubuntu:
docker pull zabbix/zabbix-proxy-sqlite3:ubuntu-latest

#### Najnowsza wersja rozwojowa (nightly) Zabbix serwer (MySQL) na Ubuntu:
docker pull zabbix/zabbix-server-mysql:ubuntu-trunk

#### Najnowsze wydanie poboczne 8.0 Zabbix serwer (MySQL) na Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-8.0-latest
```

```
#### Wersja 8.0.1 Zabbix serwer (MySQL) na Alpine Linux:
docker pull zabbix/zabbix-server-mysql:alpine-8.0.1
```

Docker Compose

Docker Compose to najszybszy sposób wdrożenia Zabbix. Odczytuje plik konfiguracyjny (plik Compose) i obsługuje całą konfigurację — pobiera obrazy Docker, tworzy wewnętrzną sieć między kontenerami, konfiguruje pamięć masową, inicjalizuje bazę danych i uruchamia wszystko we właściwej kolejności.

Repozytorium oficjalnych plików Dockerfile Zabbix zawiera gotowe do użycia pliki Docker Compose oraz system konfiguracji oparty na pliku `.env`, obsługujący różne systemy operacyjne, backendy baz danych i konfigurację komponentów Zabbix.

1. Sklonuj repozytorium, przejdź do jego katalogu i przełącz się na wersję 8.0:

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

2. (Opcjonalnie) Dostosuj wdrożenie za pomocą zmiennych środowiskowych. Ten krok można pominąć w przypadku konfiguracji domyślnej.

3. Wdróż konfigurację domyślną — serwer Zabbix, interfejs webowy (działający na NGINX) oraz backend bazy danych (MySQL lub PostgreSQL), z których każdy działa we własnym kontenerze w systemie Alpine Linux.

```
#### Z bazą danych MySQL:
docker compose -f ./compose.yaml up -d

#### Z bazą danych PostgreSQL:
docker compose -f ./compose_pgsql.yaml up -d
```

Po uruchomieniu kontenerów (zwykle w ciągu 1–3 minut) serwer Zabbix zostanie uruchomiony, a interfejs webowy będzie dostępny do logowania.

> **Note:**
> Użyj polecenia `docker compose ps`, aby sprawdzić stan kontenerów. Wszystkie kontenery (z wyjątkiem kontenera `zabbix-docker-server-db-init-1`) powinny mieć stan `Up`. Jeśli którykolwiek z nich ma stan `Exited`, sprawdź dzienniki pod kątem błędów za pomocą polecenia `docker logs -f <container-name>`.

Zmienne środowiskowe

Zachowanie Docker Compose oraz konfigurację komponentów Zabbix można dostosować za pomocą zmiennych środowiskowych.

Zmienne na poziomie Compose (zdefiniowane w pliku `.env`) określają, które obrazy Docker, porty i zakresy adresów IP sieci mają być używane. Można używać tych zmiennych w wierszu poleceń przed poleceniem `docker compose` albo edytować plik `.env`.

Na przykład poniższe polecenie wdraża kompletną konfigurację Zabbix składającą się z wielu kontenerów, używając obrazu opartego na Ubuntu (zamiast domyślnego Alpine Linux) oraz udostępniając frontend za pośrednictwem NGINX na niestandardowych portach HTTP (8282) i HTTPS (8443):

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

Zmienne na poziomie komponentu (zdefiniowane w plikach `env_vars/.env_<component>`) sterują konfiguracją komponentów Zabbix. Przed uruchomieniem polecenia `docker compose` należy edytować odpowiedni plik `.env_<component>`.

Na przykład można dostosować liczbę pasywnych procesów odpytywania serwera Zabbix i ustawienia pamięci podręcznej, a także skonfigurować strefę czasową frontendu, edytując następujące zmienne:

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

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

Note:
Zmienne środowiskowe na poziomie komponentu odpowiadają parametrom konfiguracji komponentów Zabbix, ale używają innego stylu nazewnictwa (np. `ZBX_STARTPOLLERS` odpowiada `StartPollers`). Niektóre zmienne są specyficzne dla Dockera, a niektórych parametrów konfiguracji nie można zmienić (np. `PIDFile` i `LogType`). W przypadku korzystania ze zmiennych środowiskowych należy zapoznać się z sekcją *Environment variables* w opisie obrazu Docker każdego komponentu na Docker Hub.

Wolumeny

Docker Compose przechowuje trwałe dane w katalogu `zbx_env/` utworzonym obok pliku Compose. Ten katalog zachowuje dane komponentów między restartami i aktualizacjami kontenerów.

Zawartość `zbx_env/` jest wstępnie zdefiniowana dla obrazu każdego komponentu. Na przykład:

- `zbx_env/etc/` — używany przez kontener interfejsu webowego Zabbix
- `zbx_env/usr/` i `zbx_env/var/` — używane przez kontener serwera Zabbix

Więcej informacji o każdym wolumenie można znaleźć w sekcji *Allowed volumes* w opisie obrazu Docker każdego komponentu w Docker Hub.

Makefile

Repozytorium Zabbix Dockerfiles udostępnia również plik Makefile, który stanowi skrót dla typowych zadań Docker Compose. Zamiast używać pełnego polecenia `docker compose`, można używać krótszych poleceń `make` (uruchom `make help`, aby wyświetlić dostępne opcje):

```
#### Wdrożenie domyślne (serwer Zabbix, frontend, MySQL, wszystko w systemie Alpine Linux)
make up

#### Wdrożenie niestandardowe (serwer Zabbix, frontend z niestandardowymi portami NGINX, PostgreSQL, wszys
make up \
  OS=ubuntu \
  DB=pgsql \
  ZABBIX_WEB_NGINX_HTTP_PORT=8282 \
  ZABBIX_WEB_NGINX_HTTPS_PORT=8443
```

Attention:
Podczas zatrzymywania lub usuwania kontenerów zawsze określ ten sam typ bazy danych, który został wdrożony (np. `make down DB=pgsql`).

Domyślnie polecenie `make up` uruchamia tylko minimalny zestaw usług (serwer Zabbix, frontend i bazę danych). Dzięki temu konfiguracja pozostaje lekka i nie są uruchamiane komponenty, które mogą nie być potrzebne. Aby uwzględnić dodatkowe komponenty, można użyć następujących profili Compose:

```
#### Wdrożenie domyślne + agent Zabbix, Java gateway, usługa web i pułapki SNMP:
make up COMPOSE_PROFILES=full

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

Docker (ręczne wdrożenie)

Użyj ręcznego wdrożenia, gdy chcesz wdrażać Zabbix krok po kroku, uruchamiać poszczególne komponenty, integrować je z istniejącym środowiskiem lub korzystać z alternatywnego środowiska uruchomieniowego kontenerów, takiego jak Podman.

Na przykład, aby wdrożyć proxy Zabbix, uruchom następujące polecenie:

```
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
```

To polecenie:

- Pobiera obraz Docker `zabbix/zabbix-proxy-sqlite3:alpine-8.0-latest` Docker image.
- Tworzy i uruchamia kontener `zabbix-proxy-sqlite3` oparty na obrazie Docker (z flagami `--init` i `-d`).
- Konfiguruje parametr Zabbix proxy `Server` za pomocą zmiennej środowiskowej `ZBX_SERVER_HOST`. Ten parametr określa adres IP serwera Zabbix, z którego proxy będzie pobierać dane konfiguracyjne i do którego będzie wysyłać zebrane dane. Zwróć uwagę, że inne zmienne wymagane do działania aktywnego proxy używają wartości domyślnych i można je pominąć.

Note:
Zmienne środowiskowe odpowiadają parametrom konfiguracji komponentów Zabbix, ale używają innego stylu nazewnictwa (np. `ZBX_SERVER_HOST` odpowiada `Server`). Niektóre zmienne są specyficzne dla Dockera, a niektórych parametrów konfiguracyjnych nie można zmienić (np. `PIDFile` i `LogType`). Podczas korzystania ze zmiennych środowiskowych zapoznaj się z sekcją *Environment variables* w opisie Docker image każdego komponentu na Docker Hub.

- Podłącza lokalizację magazynu zarządzaną przez Docker (np. `/var/lib/docker/volumes/zabbix-proxy-data/`) do katalogu `/var/lib/zabbix` kontenera, aby dane proxy Zabbix były przechowywane trwale, nawet jeśli kontener zostanie usunięty.

Note:
Więcej informacji o każdym woluminie znajdziesz w sekcji *Allowed volumes* w opisie Docker image każdego komponentu na Docker Hub.

Po wdrożeniu kontenera proxy Zabbix możesz przejść do dodania proxy w interfejsie web frontend Zabbix oraz skonfigurowania hostów, które mają być monitorowane przez to proxy.

Poniższe przykłady obejmują trzy dodatkowe scenariusze wdrożenia:

- serwer Zabbix (MySQL) z Java gateway
- serwer Zabbix (PostgreSQL) z pułapkami SNMP
- serwer Zabbix (MySQL) z Java gateway na RHEL 8–10

Więcej przykładów znajdziesz w opisie Docker image każdego komponentu na Docker Hub.

Wdrażanie serwera Zabbix (MySQL) z Java gateway

Ten przykład pokazuje, jak wdrożyć serwer Zabbix z bazą danych MySQL, Java gateway do monitorowania JMX oraz interfejsem webowym opartym na NGINX.

1. Utwórz dedykowaną sieć Docker, aby wszystkie kontenery komponentów Zabbix mogły komunikować się ze sobą za pomocą nazw kontenerów:

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

2. Uruchom pusty kontener serwera MySQL (zastąp `zabbix_pwd` i `root_pwd` silnymi hasłami i użyj tych wartości w kolejnych krokach):

```
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. Śledź logi MySQL i poczekaj, aż MySQL będzie gotowy na połączenia (użyj `Ctrl+C`, aby wyjść z logów):

```
docker logs -f mysql-server

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

4. Uruchom kontener Java gateway Zabbix:

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

5. Włącz `log_bin_trust_function_creators` w kontenerze serwera MySQL. Tymczasowo zmniejsza to ograniczenia bezpieczeństwa dotyczące funkcji składowanych; ustawienie zostanie wyłączone w kolejnych krokach. Umożliwia użytkownikowi bazy danych Zabbix tworzenie funkcji składowanych bez uprawnienia SUPER, wymaganego przez MySQL, gdy włączone jest logowanie binarne (domyślnie od MySQL 8.0). Uruchom następujące polecenie:

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

6. Uruchom kontener serwera Zabbix i połącz go z kontenerami Java Gateway oraz serwera MySQL:

```
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. Śledź logi serwera Zabbix i poczekaj, aż serwer Zabbix zakończy inicjalizację schematu bazy danych (użyj `Ctrl+C`, aby wyjść z logów):

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

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

8. Wyłącz `log_bin_trust_function_creators` w kontenerze serwera MySQL. Ponownie włącza to ograniczenie bezpieczeństwa, które uniemożliwia użytkownikom bez uprawnienia SUPER tworzenie funkcji składowanych. Uruchom następujące polecenie:

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

9. Uruchom kontener frontend Zabbix i połącz go z kontenerami serwera Zabbix oraz serwera MySQL:

```
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
```

Po uruchomieniu kontenerów serwer Zabbix rozpocznie działanie, a frontend będzie dostępny do logowania.

> **Note:**
> Użyj `docker ps`, aby sprawdzić stan kontenerów. Wszystkie kontenery powinny mieć stan `Up`. Jeśli którykolwiek ma stan `Exited`, sprawdź jego logi pod kątem błędów za pomocą `docker logs -f <container-name>`.

Wdrażanie serwera Zabbix (PostgreSQL) z pułapkami SNMP

Ten przykład pokazuje, jak wdrożyć serwer Zabbix z bazą danych PostgreSQL, pułapkami SNMP oraz interfejsem webowym opartym na NGINX.

1. Utwórz dedykowaną sieć Docker, aby wszystkie kontenery komponentów Zabbix mogły komunikować się ze sobą za pomocą nazw kontenerów:

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

2. Uruchom pusty kontener serwera PostgreSQL (zastąp `zabbix_pwd` silnym hasłem i użyj tej wartości w kolejnych krokach):

```
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. Uruchom kontener pułapek SNMP 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. Uruchom kontener serwera Zabbix i połącz go z kontenerami serwera PostgreSQL oraz pułapek 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. Śledź logi serwera Zabbix i poczekaj, aż serwer Zabbix zakończy inicjalizację schematu bazy danych (użyj `Ctrl+C`, aby wyjść z logów):

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

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

6. Uruchom kontener frontend Zabbix i połącz go z kontenerami serwera Zabbix oraz serwera PostgreSQL:

```
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
```

Po uruchomieniu kontenerów serwer Zabbix rozpocznie działanie, a frontend będzie dostępny do logowania.

> **Note:**
> Użyj `docker ps`, aby sprawdzić stan kontenerów. Wszystkie kontenery powinny mieć stan Up. Jeśli którykolwiek ma stan Exited, sprawdź jego logi pod kątem błędów za pomocą `docker logs -f <container-name>`.

Wdrażanie serwera Zabbix (MySQL) z Java gateway w systemie RHEL 8-10

Ten przykład pokazuje, jak wdrożyć serwer Zabbix z bazą danych MySQL, Java gateway do monitorowania JMX oraz interfejsem

webowym opartym na NGINX, działające w systemie Red Hat Enterprise Linux 8, 9 lub 10.

W systemie Red Hat Enterprise Linux zalecanym środowiskiem uruchomieniowym kontenerów jest Podman zamiast Docker. Podman działa podobnie do Docker, ale nie wymaga działającej w tle usługi uruchomionej jako root, dzięki czemu lepiej sprawdza się w środowiskach Red Hat.

1. Utwórz nowy pod o nazwie `zabbix` z udostępnionymi portami dla frontendu webowego Zabbix i trappera serwera Zabbix:

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

2. (Opcjonalnie) Uruchom kontener agenta Zabbix w podzie `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. Utwórz katalog `./mysql/` na hoście RHEL:

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

4. Uruchom pusty kontener serwera MySQL (zastąp `zabbix_pwd` i `root_pwd` silnymi hasłami i użyj tych wartości w kolejnych krokach):

```
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. Śledź logi MySQL i poczekaj, aż MySQL będzie gotowy na połączenia (użyj `Ctrl+C`, aby wyjść z logów):

```
podman logs -f mysql-server

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

6. Uruchom kontener Java gateway Zabbix:

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

7. Włącz `log_bin_trust_function_creators` w kontenerze serwera MySQL. Tymczasowo zmniejsza to ograniczenia bezpieczeństwa dotyczące funkcji składowanych; ustawienie zostanie wyłączone w kolejnych krokach. Umożliwia użytkownikowi bazy danych Zabbix tworzenie funkcji składowanych bez uprawnienia `SUPER`, wymaganego przez MySQL, gdy włączone jest logowanie binarne (domyślnie od MySQL 8.0). Uruchom następujące polecenie:

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

8. Uruchom kontener serwera Zabbix i połącz go z kontenerami Java Gateway oraz serwera MySQL:

```
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. Śledź logi serwera Zabbix i poczekaj, aż serwer Zabbix zakończy inicjalizację schematu bazy danych (użyj Ctrl+C, aby wyjść z logów):

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

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

10. Wyłącz log_bin_trust_function_creators w kontenerze serwera MySQL. Ponownie włącza to ograniczenie bezpieczeństwa, które uniemożliwia użytkownikom bez uprawnienia SUPER tworzenie funkcji składowanych. Uruchom następujące polecenie:

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

11. Uruchom kontener frontend Zabbix i połącz go z kontenerami serwera Zabbix oraz serwera MySQL:

```
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:**
> Pod zabbix udostępnia hostowi port 80/TCP (HTTP), mapując go z portu 8080/TCP kontenera zabbix-web-mysql.

Po uruchomieniu kontenerów serwer Zabbix rozpocznie działanie, a frontend będzie dostępny do logowania.

> **Note:**
> Użyj docker ps, aby sprawdzić stan kontenerów. Wszystkie kontenery powinny mieć stan Up. Jeśli którykolwiek ma stan Exited, sprawdź jego logi pod kątem błędów za pomocą docker logs -f <container-name>.

Instalacja ze źródeł

Najnowszą wersję Zabbix można uzyskać, kompilując ją ze źródeł. Zobacz także Pobieranie kodu źródłowego Zabbix.

Tutaj znajduje się samouczek krok po kroku dotyczący instalacji Zabbix ze źródeł.

Instalowanie demonów Zabbix

1 Pobierz archiwum źródłowe

Przejdź do strony pobierania Zabbix i pobierz archiwum źródłowe. Po pobraniu rozpakuj źródła, uruchamiając:

```
tar -zxvf zabbix-8.0.0.tar.gz
```

> **Note:**
> Wprowadź w poleceniu prawidłową wersję Zabbix. Musi ona odpowiadać nazwie pobranego archiwum.

2 Utwórz konto użytkownika

Wszystkie procesy demonów Zabbix działają pod nieuprzywilejowanymi użytkownikami systemowymi. Jeśli demon Zabbix zostanie uruchomiony z nieuprzywilejowanego konta użytkownika, będzie nadal działał jako ten użytkownik.

W domyślnej konfiguracji, jeśli demon zostanie uruchomiony jako root, przełączy się na konto użytkownika zabbix, które musi istnieć. Aby utworzyć użytkownika systemowego zabbix i grupę, uruchom poniższe polecenia.

System oparty na RedHat:

```
groupadd --system zabbix
useradd --system -g zabbix -d /usr/lib/zabbix -s /sbin/nologin -c "Zabbix Monitoring System" zabbix
```

System oparty na Debianie:

```
addgroup --system --quiet zabbix
adduser --quiet --system --disabled-login --ingroup zabbix --home /var/lib/zabbix --no-create-home zabbix
```

Nie ma potrzeby tworzenia osobnego konta użytkownika dla frontend Zabbix.

Zalecenie dotyczące bezpieczeństwa

Jeśli Zabbix serwer i agent działają na tej samej maszynie, zaleca się uruchamianie ich pod **oddzielnymi kontami użytkowników**. Uruchamianie obu pod tym samym użytkownikiem pozwala agentowi uzyskać dostęp do pliku konfiguracyjnego serwera, co może ujawnić poufne informacje — takie jak hasło do bazy danych — każdemu użytkownikowi z uprawnieniami administratora w Zabbix.

> **Attention:**
> Uruchamianie Zabbix jako `root`, `bin` lub dowolnego innego konta ze specjalnymi uprawnieniami stanowi zagrożenie bezpieczeństwa.

Katalog domowy (opcjonalnie)

Procesy Zabbix nie wymagają katalogu domowego, dlatego jego tworzenie zasadniczo nie jest zalecane. Jeśli jednak potrzebujesz funkcjonalności wymagającej katalogu domowego (na przykład przechowywania poświadczeń MySQL w $HOME/.my.cnf), możesz go utworzyć za pomocą poniższych poleceń.

W systemach opartych na RedHat uruchom:

```
mkdir -m u=rwx,g=rwx,o= -p /usr/lib/zabbix
chown zabbix:zabbix /usr/lib/zabbix
```

W systemach opartych na Debianie uruchom:

```
mkdir -m u=rwx,g=rwx,o= -p /var/lib/zabbix
chown zabbix:zabbix /var/lib/zabbix
```

3 Tworzenie bazy danych Zabbix

Dla demonów Zabbix serwer i proxy, a także dla frontendu Zabbix, wymagana jest baza danych. Nie jest ona potrzebna do uruchomienia Zabbix agent.

Dostępne są skrypty SQL do utworzenia schematu bazy danych i wstawienia zestawu danych. Baza danych proxy Zabbix wymaga tylko schematu, natomiast baza danych serwera Zabbix wymaga oprócz schematu również zestawu danych.

Po utworzeniu bazy danych Zabbix przejdź do kolejnych kroków kompilacji Zabbix.

4 Konfiguracja źródeł

Do zbudowania serwera Zabbix, proxy Zabbix lub agent wymagany jest standard C99 z rozszerzeniami GNU. Wersję tę można jawnie określić, ustawiając CFLAGS="-std=gnu99":

```
export CFLAGS="-std=gnu99"
```

> **Note:**
> W przypadku instalacji z repozytorium Git Zabbix najpierw należy uruchomić:
> `./bootstrap.sh`

Podczas konfigurowania źródeł serwera Zabbix lub proxy należy określić typ używanej bazy danych. W danym momencie z procesem serwera lub proxy można skompilować tylko jeden typ bazy danych.

Aby wyświetlić wszystkie obsługiwane opcje konfiguracji, uruchom w rozpakowanym katalogu źródłowym Zabbix:

```
./configure --help
```

Aby skonfigurować źródła serwera Zabbix i agent, możesz uruchomić polecenie podobne do:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --wit
```

Aby skonfigurować źródła serwera Zabbix (z PostgreSQL itd.), możesz uruchomić:

```
./configure --enable-server --with-postgresql --with-net-snmp
```

Aby skonfigurować źródła proxy Zabbix (z SQLite itd.), możesz uruchomić:

```
./configure --prefix=/usr --enable-proxy --with-net-snmp --with-sqlite3 --with-ssh2
```

Aby skonfigurować źródła agent, możesz uruchomić:

```
./configure --enable-agent
```

lub w przypadku agent 2:

```
./configure --enable-agent2
```

Note:
Do zbudowania agent 2 należy zainstalować obsługiwaną wersję Go.

Uwagi dotyczące opcji kompilacji:

- `--enable-agent` - kompiluje agent, a także narzędzia wiersza poleceń Zabbix get i Zabbix sender.
- `--with-libcurl` - wymagane do monitorowania maszyn wirtualnych, uwierzytelniania SMTP oraz pozycji `web.page.*` agent Zabbix i pozycji zapytania telemetrycznego. Zobacz także: Wymagania (libcurl).
- `--with-libxml2` - wymagane do monitorowania maszyn wirtualnych.
- `--with-libpcre2[=DIR]` - Zabbix zawsze kompiluje się z biblioteką PCRE2; ta opcja umożliwia jedynie określenie niestandardowej ścieżki instalacji PCRE2.
- `--with-mysql=/path/to/mysql_config` - określa ścieżkę do konfiguracji konkretnej biblioteki klienta MySQL. Jest przydatne, gdy zainstalowanych jest wiele wersji MySQL lub MariaDB.
- `--enable-static` - statycznie łączy biblioteki (nie jest obsługiwane w systemie Solaris). Użyj tej opcji, jeśli planujesz dystrybuować skompilowane pliki binarne do systemów bez wymaganych bibliotek. Nie jest zalecane podczas budowania serwera Zabbix. Aby zbudować serwer statycznie, wymagana jest statyczna wersja każdej biblioteki zewnętrznej. Skrypt configure nie sprawdza tego automatycznie.
- `--with-stacksize=<value>` - ustawia rozmiar stosu dla każdego wątku w kilobajtach (np. `--with-stacksize=512`). Możesz zwiększyć tę wartość, jeśli Zabbix ulega awarii lub zawiesza się z powodu przepełnienia stosu (np. podczas wstępnego przetwarzania w systemach z niskimi domyślnymi limitami stosu wątków).
- `--with-apm` (używaj tylko z `--enable-proxy`) - wymagane do monitorowania wydajności aplikacji (APM) na proxy Zabbix. Zobacz także: Wymagania (g++, gRPC, protobuf, Abseil).

Attention:
Jeśli `./configure` zakończy się niepowodzeniem z powodu brakujących bibliotek lub innych problemów, sprawdź plik `config.log`, aby uzyskać szczegółowe informacje o błędzie.
Na przykład jeśli brakuje `libssl`, bezpośredni komunikat o błędzie może być mylący:
```
checking for main in -lmysqlclient... no
configure: error: Not found mysqlclient library
```
W takim przypadku plik `config.log` ujawnia rzeczywistą przyczynę:
```
/usr/bin/ld: cannot find -lssl
/usr/bin/ld: cannot find -lcrypto
```

Zobacz także:

- Kompilowanie Zabbix z obsługą szyfrowania
- Znane problemy z kompilacją.

5 Zbuduj i zainstaluj wszystko

Note:
Jeśli instalujesz z repozytorium Git Zabbix, najpierw należy uruchomić:
```
$ make dbschema
```

```
make install
```

Ten krok należy wykonać jako użytkownik z wystarczającymi uprawnieniami (zwykle `root` lub przy użyciu `sudo`).

Uruchomienie `make install` domyślnie zainstaluje pliki binarne demona (zabbix_server, zabbix_agentd, zabbix_proxy) w /usr/local/sbin oraz pliki binarne klienta (zabbix_get, zabbix_sender) w /usr/local/bin.

Note:
Aby wskazać inną lokalizację niż `/usr/local`, użyj klucza `--prefix` w poprzednim kroku konfiguracji źródeł, na przykład `--prefix=/home/zabbix`. W takim przypadku pliki binarne demona zostaną zainstalowane w `<prefix>/sbin`, natomiast narzędzia w `<prefix>/bin`. Strony podręcznika zostaną zainstalowane w `<prefix>/share`.

6 Przeglądanie i edytowanie plików konfiguracyjnych

- edytuj plik konfiguracyjny agenta Zabbix (`zabbix_agentd.conf`).

Ten plik należy skonfigurować dla każdego hosta, na którym zainstalowano `zabbix_agentd`.

W pliku należy określić **adres IP** serwera Zabbix. Połączenia z innych hostów będą odrzucane.

- edytuj plik konfiguracyjny serwera Zabbix (`zabbix_server.conf`).

Należy określić nazwę bazy danych, użytkownika i hasło (jeśli są używane).

Pozostałe parametry będą odpowiednie z wartościami domyślnymi, jeśli masz małą instalację (do dziesięciu monitorowanych hostów). Jeśli jednak chcesz zmaksymalizować wydajność serwera Zabbix (lub proxy), zmień parametry domyślne.

- jeśli zainstalowano proxy Zabbix, edytuj plik konfiguracyjny proxy (`zabbix_proxy.conf`).

Należy określić adres IP serwera i nazwę hosta proxy (musi być znana serwerowi), a także nazwę bazy danych, użytkownika i hasło (jeśli są używane).

> **Note:**
> W przypadku SQLite należy określić pełną ścieżkę do pliku bazy danych; użytkownik i hasło do bazy danych nie są wymagane.

7 Uruchom demony

Uruchom `zabbix_server` po stronie serwera.

```
zabbix_server
```

> **Note:**
> Upewnij się, że system pozwala na przydzielenie 36 MB (lub nieco więcej) pamięci współdzielonej, w przeciwnym razie serwer może się nie uruchomić i w pliku dziennika serwera zobaczysz komunikat „Cannot allocate shared memory for <type of cache>.”. Może się to zdarzyć w systemach FreeBSD i Solaris 8.

Uruchom `zabbix_agentd` na wszystkich monitorowanych maszynach.

```
zabbix_agentd
```

> **Note:**
> Upewnij się, że system pozwala na przydzielenie 2 MB pamięci współdzielonej, w przeciwnym razie agent może się nie uruchomić i w pliku dziennika agenta zobaczysz komunikat „Cannot allocate shared memory for collector.”. Może się to zdarzyć w systemie Solaris 8.

Jeśli zainstalowano Zabbix proxy, uruchom `zabbix_proxy`.

```
zabbix_proxy
```

Instalacja interfejsu webowego Zabbix

Kopiowanie plików PHP

frontend Zabbix jest napisany w PHP, więc do jego uruchomienia potrzebny jest serwer WWW obsługujący PHP. Instalacja polega po prostu na skopiowaniu plików PHP z katalogu ui do katalogu dokumentów HTML serwera WWW.

Typowe lokalizacje katalogów dokumentów HTML dla serwerów Apache obejmują:

- /usr/local/apache2/htdocs (domyślny katalog podczas instalacji Apache ze źródeł)
- /srv/www/htdocs (OpenSUSE, SLES)
- /var/www/html (Debian, Ubuntu, Fedora, RHEL)

Zaleca się użycie podkatalogu zamiast katalogu głównego HTML. Aby utworzyć podkatalog i skopiować pliki frontend Zabbix do niego, wykonaj następujące polecenia, zastępując <htdocs> rzeczywistym katalogiem:

```
mkdir <htdocs>/zabbix
cd ui
cp -a . <htdocs>/zabbix
```

Jeśli planujesz używać innego języka niż angielski, zobacz Instalacja dodatkowych języków frontend, aby uzyskać instrukcje.

Instalacja frontend

Aby uzyskać informacje o kreatorze instalacji frontend Zabbix, zobacz stronę Instalacja interfejsu webowego.

Instalowanie ładowalnych wtyczek Zabbix agent 2

Instalacja ładowalnych wtyczek Zabbix agent 2 jest wymagana tylko wtedy, gdy chcesz monitorować cele nieobjęte przez wbudowane wtyczki (np. serwery lub klastry MongoDB, PostgreSQL i jego forki itp.). Zobacz pełną listę ładowalnych wtyczek oraz wbudowanych wtyczek.

Attention:
Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać szczegółowe wymagania i instrukcje instalacji.

Aby zainstalować ze źródeł, najpierw pobierz i rozpakuj archiwum źródłowe ładowalnej wtyczki.

Aby skompilować wtyczkę, przejdź do rozpakowanego katalogu wtyczki i uruchom `make`:

```
make
```

Note:
Do zbudowania ładowalnych wtyczek Zabbix agent 2 musi być zainstalowana obsługiwana wersja Go.

Plik wykonywalny wtyczki może znajdować się w dowolnym miejscu, o ile może być załadowany przez Zabbix agent 2. Określ ścieżkę do binarnego pliku wtyczki w pliku konfiguracyjnym wtyczki, np. w `postgresql.conf` dla wtyczki PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

Ścieżka do pliku konfiguracyjnego wtyczki musi być określona w parametrze `Include` pliku konfiguracyjnego Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Przejdź do sekcji konfiguracja, aby uzyskać więcej informacji na temat konfigurowania wtyczek.

Ładowalne wtyczki dostarczane przez Zabbix używają prostych plików makefile z następującymi celami kompilacji:

- `make` - zbuduj wtyczkę
- `make clean` - usuń wszystkie pliki utworzone podczas budowania wtyczki
- `make check` - uruchom testy własne (wymaga rzeczywistego celu monitorowania, np. bazy danych PostgreSQL)
- `make style` - sprawdź styl kodu Go za pomocą `golangci-lint`
- `make format` - sformatuj kod Go za pomocą `go fmt`
- `make dist` - utwórz archiwum źródłowe zawierające wszystkie zależności

Instalacja Java gateway

Instalacja Java gateway jest wymagana tylko wtedy, gdy chcesz monitorować aplikacje JMX. Java gateway jest lekki i nie wymaga bazy danych.

Aby zainstalować ze źródeł, najpierw pobierz i rozpakuj archiwum źródłowe.

Aby skompilować Java gateway, uruchom skrypt `./configure` z opcją `--enable-java`. Zaleca się określenie opcji `--prefix`, aby wskazać ścieżkę instalacji inną niż domyślna /usr/local, ponieważ instalacja Java gateway utworzy całe drzewo katalogów, a nie tylko jeden plik wykonywalny.

```
./configure --enable-java --prefix=$PREFIX
```

Aby skompilować i spakować Java gateway do pliku JAR, uruchom `make`. Pamiętaj, że na tym etapie w ścieżce musisz mieć pliki wykonywalne `javac` i `jar`.

```
make
```

Teraz masz plik zabbix-java-gateway-$VERSION.jar w katalogu `src/zabbix_java/bin`. Jeśli chcesz uruchamiać Java gateway z katalogu `src/zabbix_java` w katalogu dystrybucyjnym, możesz przejść do instrukcji konfiguracji i uruchamiania Java gateway. W przeciwnym razie upewnij się, że masz odpowiednie uprawnienia i uruchom `make install`.

```
make install
```

Przejdź do konfiguracji, aby uzyskać więcej informacji na temat konfigurowania i uruchamiania Java gateway.

Instalacja usługi webowej Zabbix

Instalacja usługi webowej Zabbix jest wymagana tylko wtedy, gdy chcesz używać raportów zaplanowanych.

Aby zainstalować ze źródeł, najpierw pobierz i rozpakuj archiwum źródłowe.

Aby skompilować usługę webową Zabbix, uruchom skrypt `./configure` z opcją `--enable-webservice`.

Note:
Do zbudowania usługi webowej Zabbix musi być zainstalowana obsługiwana wersja Go.

Uruchom zabbix_web_service na maszynie, na której zainstalowana jest usługa webowa:

```
zabbix_web_service
```

Przejdź do konfiguracji, aby uzyskać więcej informacji na temat konfiguracji generowania raportów zaplanowanych.

Instalowanie serwera Zabbix MCP

Aby zainstalować z kodu źródłowego, najpierw pobierz i rozpakuj archiwum źródłowe.

Aby skompilować serwer Zabbix MCP, uruchom następujące polecenie w katalogu kodu źródłowego serwera MCP:

```
make build
```

> **Note:**
> Do zbudowania serwera Zabbix MCP musi być zainstalowana obsługiwana wersja Go.

Aby zgłaszać określoną wersję w danych wyjściowych polecenia `zabbix_mcp_server -V` oraz w nagłówku `User-Agent` żądań API Zabbix, ustaw zmienną `VERSION`, na przykład `make build VERSION=8.0.0`.

Aby zainstalować plik binarny w katalogu `/usr/local/sbin`, a przykładowy plik konfiguracyjny w katalogu `/etc/zabbix`, uruchom:

```
sudo make install
```

Edytuj plik konfiguracyjny `/etc/zabbix/zabbix_mcp_server.conf` i ustaw co najmniej następujące parametry:

- `FrontendURL` - adres URL API Zabbix zakończony ciągiem `api_jsonrpc.php`;
- `AllowedIP` - adresy klientów MCP, którym zezwolono na nawiązywanie połączeń.

Aby szyfrować połączenia od klientów MCP, ustaw `TLSAccept=cert` i określ `TLSCertFile` oraz `TLSKeyFile` albo uruchom serwer MCP za zaufanym odwrotnym proxy obsługującym TLS.

Jeśli użytkownik `zabbix` nie istnieje (na przykład gdy na komputerze zainstalowano tylko serwer MCP), utwórz go zgodnie z opisem w sekcji Tworzenie konta użytkownika.

Sprawdź plik konfiguracyjny i uruchom zabbix_mcp_server:

```
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf -T
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf
```

Przejdź do sekcji Serwer MCP, aby uzyskać więcej informacji na temat konfigurowania serwera i łączenia klientów MCP.

Pobieranie kodu źródłowego Zabbix

Istnieje kilka sposobów pobrania kodu źródłowego Zabbix:

- Możesz pobrać wydane stabilne wersje z oficjalnej strony internetowej Zabbix
- Możesz pobrać kompilacje nightly ze strony dla deweloperów w oficjalnym serwisie Zabbix
- Możesz pobrać najnowszą wersję rozwojową z repozytorium kodu źródłowego Git:
  - Główna lokalizacja pełnego repozytorium znajduje się pod adresem https://git.zabbix.com/scm/zbx/zabbix.git
  - Gałąź master oraz obsługiwane wydania są również mirrorowane w Github pod adresem https://github.com/zabbix/zabbix

Aby sklonować repozytorium, musi być zainstalowany klient Git. Oficjalny pakiet klienta Git uruchamianego z wiersza poleceń jest w dystrybucjach zwykle nazywany **git**. Aby go zainstalować, na przykład w Debian/Ubuntu, uruchom:

```
sudo apt-get update
sudo apt-get install git
```

Aby pobrać cały kod źródłowy Zabbix, przejdź do katalogu, w którym chcesz umieścić kod, i wykonaj:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

Problemy z kompilacją

Są to znane problemy dotyczące kompilacji Zabbix ze źródeł. We wszystkich pozostałych przypadkach zobacz stronę Znane problemy.

Biblioteka w niestandardowej lokalizacji

Zabbix umożliwia wskazanie biblioteki znajdującej się w niestandardowej lokalizacji. W poniższym przykładzie Zabbix uruchomi `curl-config` z podanej niestandardowej lokalizacji i użyje jego danych wyjściowych do określenia właściwej biblioteki libcurl.

```
$ ./configure --enable-server --with-mysql --with-libcurl=/usr/local/bin/curl-config
```

To zadziała, jeśli jest to jedyna biblioteka libcurl zainstalowana w systemie, ale może nie zadziałać, jeśli inna biblioteka libcurl jest zainstalowana w standardowej lokalizacji (na przykład przez menedżera pakietów). Taka sytuacja występuje wtedy, gdy dla Zabbix potrzebna jest nowsza wersja biblioteki, a starsza dla innych aplikacji.

Dlatego wskazanie komponentu w niestandardowej lokalizacji nie zawsze będzie działać, jeśli ten sam komponent istnieje również w standardowej lokalizacji.

Na przykład, jeśli używasz nowszej biblioteki libcurl zainstalowanej w `/usr/local`, podczas gdy pakiet libcurl nadal jest zainstalowany, Zabbix może wybrać niewłaściwą bibliotekę i kompilacja zakończy się niepowodzeniem:

```
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
```

Tutaj funkcja `curl_easy_header()` nie jest dostępna w starszej bibliotece `/usr/lib/x86_64-linux-gnu/libcurl.so`, ale jest dostępna w nowszej `/usr/local/lib/libcurl.so`.

Problem wynika z kolejności flag linkera, a jednym z rozwiązań jest podanie pełnej ścieżki do biblioteki w zmiennej LDFLAGS:

```
$ LDFLAGS="-Wl,--no-as-needed /usr/local/lib/libcurl.so" ./configure --enable-server --with-mysql --with-l
```

Zwróć uwagę na opcję `-Wl,--no-as-needed`, która może być wymagana w niektórych systemach (zobacz także: domyślne opcje linkowania w systemach opartych na Debianie).

Zbyt mały rozmiar stosu na niektórych systemach

Jeśli Zabbix ulega awarii lub zawiesza się z powodu przepełnienia stosu, możesz zwiększyć rozmiar stosu dla pojedynczego wątku, używając opcji `--with-stacksize` podczas konfigurowania źródeł. Problem ten może wystąpić na systemach z niskimi domyślnymi limitami stosu wątków, zwłaszcza podczas preprocessing, gdy tworzonych jest wiele wątków.

Poniższy przykład ustawia rozmiar stosu na 512 KB na wątek:

```
./configure --enable-server --with-mysql --with-stacksize=512
```

Limity stosu wątków systemu możesz sprawdzić w czasie działania za pomocą polecenia `ulimit -s` w systemach opartych na Linux.

Uwagi dotyczące instalacji z pakietów

Przegląd

Oficjalne pakiety **przedpremierowe** Zabbix 8.0 są dostępne na stronie internetowej Zabbix, gdzie można wybrać system operacyjny i komponent Zabbix, aby wygenerować instrukcje oraz polecenia instalacji.

Oprócz poniższych uwag dotyczących instalacji pakietów zapoznaj się również ze stroną Instalacja i konfiguracja.

Pakiety są dostępne dla następujących dystrybucji systemu Linux:

- Red Hat Enterprise Linux i jego pochodne: 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:**
> Niektóre dystrybucje systemu operacyjnego, w szczególności dystrybucje oparte na Debianie, udostępniają własne pakiety Zabbix. Pakiety te **nie są** obsługiwane przez Zabbix i mogą być nieaktualne lub nie zawierać najnowszych funkcji oraz poprawek błędów. Zaleca się używanie wyłącznie oficjalnych pakietów z Oficjalnego repozytorium Zabbix. Jeśli wcześniej zainstalowano Zabbix z repozytorium systemu operacyjnego, zapoznaj się z instrukcjami aktualizacji pakietów Zabbix z repozytoriów systemu operacyjnego.

Pakiety obsługują bazy danych MySQL/PostgreSQL oraz serwery internetowe Apache/Nginx. Należy pamiętać, że serwer Zabbix i proxy nie mogą współdzielić tej samej bazy danych; jeśli oba komponenty są zainstalowane na tym samym hoście, należy użyć różnych nazw baz danych.

W razie potrzeby oddzielne pakiety dla agent/agent 2 Zabbix, Zabbix get i Zabbix sender są dostępne w Oficjalnym repozytorium Zabbix.

W systemach opartych na Debianie lub Ubuntu zainstalowanie dodatkowego pakietu Zabbix za pomocą apt może również wybrać aktualizacje już zainstalowanych pakietów Zabbix z tego samego repozytorium. Aby zainstalować pakiet bez aktualizowania zainstalowanego pakietu serwera Zabbix lub proxy, tymczasowo wstrzymaj ten pakiet przed instalacją, a następnie usuń wstrzymanie:

```
apt-mark hold zabbix-server-mysql
apt install zabbix-sender
apt-mark unhold zabbix-server-mysql
```

Zabbix udostępnia również wstępnie skompilowane pliki binarne agent Zabbix dla systemów operacyjnych innych niż Linux; zobacz:

- Instalacja agent Zabbix z MSI w systemie Windows
- Instalacja agent Zabbix z PKG w systemie macOS
- Agent Zabbix w systemie IBM AIX
- Starsze pliki binarne (dla starszych lub mniej popularnych systemów, takich jak HP-UX, NetBSD, Tru64 oraz starsze wersje SLES)

Uwagi dotyczące instalacji pakietów

Poniższe uwagi dotyczą wszystkich systemów:

- W przypadku korzystania z PostgreSQL ustawienie `DBHost=localhost` (lub adresu IP) w konfiguracji serwera/proxy Zabbix powoduje używanie przez PostgreSQL gniazda sieciowego zamiast lokalnego gniazda UNIX; informacje dotyczące powiązanej konfiguracji zawiera sekcja Konfiguracja SELinux.
- W przypadku korzystania z TimescaleDB zapoznaj się z dodatkowymi informacjami dotyczącymi konfiguracji TimescaleDB.
- W przypadku instalowania Java gateway Zabbix (do monitorowania aplikacji JMX) zapoznaj się z dodatkowymi instrukcjami konfiguracji dla systemów opartych na RHEL oraz systemów opartych na Debianie.
- W przypadku instalowania serwera MCP Zabbix zainstaluj pakiet `zabbix-mcp-server` i zaktualizuj plik konfiguracyjny (co najmniej `FrontendURL` i `AllowedIP`) przed uruchomieniem usługi.
- Informacje dotyczące uruchamiania agenta Zabbix jako użytkownik root zawiera sekcja Uruchamianie agenta jako użytkownik root.
- Monitorowanie wydajności aplikacji (APM) na proxy Zabbix jest obsługiwane przez następujące pakiety:
  - RHEL 10 i jego pochodne - zainstaluj pakiet APM proxy Zabbix (np. `zabbix-proxy-mysql-apm` zamiast `zabbix-proxy-mysql`); wymaga to również zainstalowania repozytorium EPEL i włączenia repozytoriów pakietów zależności CRB.
  - Debian 12 lub nowszy, Raspberry Pi OS (Raspbian) 12 lub nowszy, Ubuntu 24.04 lub nowszy - APM jest zawarte w pakietach proxy Zabbix.
  - W innych systemach można zainstalować proxy Zabbix ze źródeł za pomocą opcji `--with-apm`.

Poniższe uwagi dotyczą systemu RHEL i jego pochodnych:

- Jeśli włączono repozytorium EPEL (np. na potrzeby pakietu APM proxy Zabbix), które również udostępnia pakiety Zabbix, należy wykluczyć je z rozwiązywania zależności pakietów przed zainstalowaniem oficjalnych pakietów Zabbix; zobacz Przypadkowa instalacja pakietów Zabbix z EPEL.
- Informacje dotyczące instalowania pakietów Zabbix w środowiskach Red Hat UBI zawiera sekcja Pakiety Zabbix dla RHEL w środowiskach Red Hat UBI.
- W przypadku korzystania z pozycji ICMP ping pakiety `fping` są również dostępne w oficjalnym repozytorium Zabbix.

Konfiguracja SELinux

Zabbix używa komunikacji międzyprocesowej opartej na gniazdach. W systemach, w których włączono Security-Enhanced Linux (SELinux), może być konieczne dodanie reguł SELinux, aby zezwolić Zabbix na tworzenie/używanie gniazd domeny UNIX w katalogu SocketDir. Pliki gniazd są używane przez serwer Zabbix (alerter, preprocessing, IPMI) oraz proxy Zabbix (IPMI) i są obecne, dopóki proces jest uruchomiony.

Przy włączonym SELinux w trybie enforcing wykonaj następujące polecenia, aby umożliwić komunikację między frontend Zabbix a serwerem:

Dla RHEL 7 (i nowszych), AlmaLinux, CentOS Stream, Oracle Linux, Rocky Linux 8 (i nowszych):

```
setsebool -P httpd_can_connect_zabbix on
```

Jeśli baza danych jest dostępna przez sieć (w tym `localhost` dla PostgreSQL), zezwól również frontend Zabbix na połączenie z bazą danych:

```
setsebool -P httpd_can_network_connect_db on
```

Dla RHEL wcześniejszych niż 7:

```
setsebool -P httpd_can_network_connect on
setsebool -P zabbix_can_network on
```

Po zastosowaniu ustawień SELinux uruchom ponownie Apache:

```
systemctl restart httpd
```

Opcjonalnie możesz zainstalować predefiniowany pakiet zabbix-selinux-policy z oficjalnego repozytorium Zabbix. Pakiet ten jest dostarczany dla wszystkich obsługiwanych wersji systemów operacyjnych, aby uprościć wdrożenie Zabbix i zapobiec wyłączaniu SELinux przez użytkowników z powodu złożoności konfiguracji.
 Dla wersji 10 AlmaLinux, CentOS Stream, RHEL, Rocky Linux i Oracle Linux polityka SELinux jest dostarczana przez pakiet zabbix-selinux-policy-main; pakiet zabbix-selinux-policy ma zależność od zabbix-selinux-policy-main lub selinux-policy-targeted-extra. Pakiet zabbix-selinux-policy-main koliduje z selinux-policy-targeted-extra; dlatego obu pakietów nie można zainstalować jednocześnie.

Attention:
Dla maksymalnego bezpieczeństwa zaleca się ustawienie niestandardowych ustawień SELinux.

Pakiet zabbix-selinux-policy (zabbix-selinux-policy-main dla wersji 10 AlmaLinux, CentOS Stream, RHEL, Rocky Linux i Oracle Linux) zawiera podstawową politykę SELinux, która umożliwia Zabbix tworzenie i używanie gniazd oraz włącza połączenie HTTPd z PostgreSQL (używanym przez frontend).

Plik źródłowy zabbix_policy.te zawiera następujące reguły:

```
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;
```

Pakiety Debuginfo

Pakiety Debuginfo zawierają symbole debugowania dla binariów Zabbix. Nie są wymagane do standardowej instalacji ani działania, ale są przydatne podczas zaawansowanego rozwiązywania problemów.

Należy pamiętać, że pakiety Debuginfo są dostępne dla następujących wersji: Red Hat Enterprise Linux 7 i wcześniejszych (w tym pochodnych) oraz SUSE Linux Enterprise Server 15 i wcześniejszych (w tym pochodnych).

Aby włączyć repozytorium zabbix-debuginfo:

- W RHEL 7 edytuj /etc/yum.repos.d/zabbix.repo i ustaw enabled=1 dla sekcji 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
```

- W SUSE edytuj /etc/zypp/repos.d/zabbix.repo i ustaw enabled=1 dla sekcji 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
```

Po włączeniu zainstaluj pakiety:

- W RHEL zainstaluj pojedynczy pakiet z informacjami debugowania dla wszystkich komponentów Zabbix:

```
dnf install zabbix-debuginfo
```

- W SUSE zainstaluj pakiety debuginfo dla poszczególnych komponentów:

```
zypper install zabbix-<component>-debuginfo
```

Włączanie repozytoriów niestabilnych wydań

Poniższe instrukcje dotyczą włączania niestabilnych repozytoriów wydań Zabbix (domyślnie wyłączonych), używanych dla kandydatów do wydania drobnych wersji Zabbix.

Najpierw zainstaluj lub zaktualizuj pakiet zabbix-release do najnowszej wersji. Aby włączyć pakiety rc w swoim systemie, wykonaj następujące czynności:

Red Hat Enterprise Linux

Otwórz plik `/etc/yum.repos.d/zabbix.repo` i ustaw `enabled=1` dla repozytorium `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

Otwórz plik `/etc/apt/sources.list.d/zabbix.list` i odkomentuj „niestabilne repozytorium Zabbix”.

```
#### 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

Otwórz plik `/etc/zypp/repos.d/zabbix.repo` i ustaw `enabled=1` dla repozytorium `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
```

Instalacja agenta Windows z pakietu MSI

Przegląd

Zabbix agent można zainstalować w systemie Windows przy użyciu 32-bitowych lub 64-bitowych pakietów instalatora MSI, dostępnych do pobrania.

Minimalne wymagania systemu operacyjnego dla instalacji MSI to:

- **Dla Zabbix agent:** Windows XP (64-bit) lub Windows Server 2003
- **Dla Zabbix agent 2:** Windows 10 (32-bit) lub Windows Server 2016

Pakietów 32-bitowych nie można instalować w systemach 64-bitowych.

Pakiety zawierają:

- obsługę TLS (konfiguracja TLS jest opcjonalna)
- narzędzia Zabbix get i Zabbix sender (mogą być instalowane razem z Zabbix agent/agent 2 lub osobno)

Attention:
Pakiety Zabbix agent 2 nie zawierają ładowalnych wtyczek (MongoDB, PostgreSQL, MSSQL), które należy pobrać i zainstalować osobno.

Instalację można przeprowadzić przy użyciu kreatora instalacji lub wiersza poleceń.

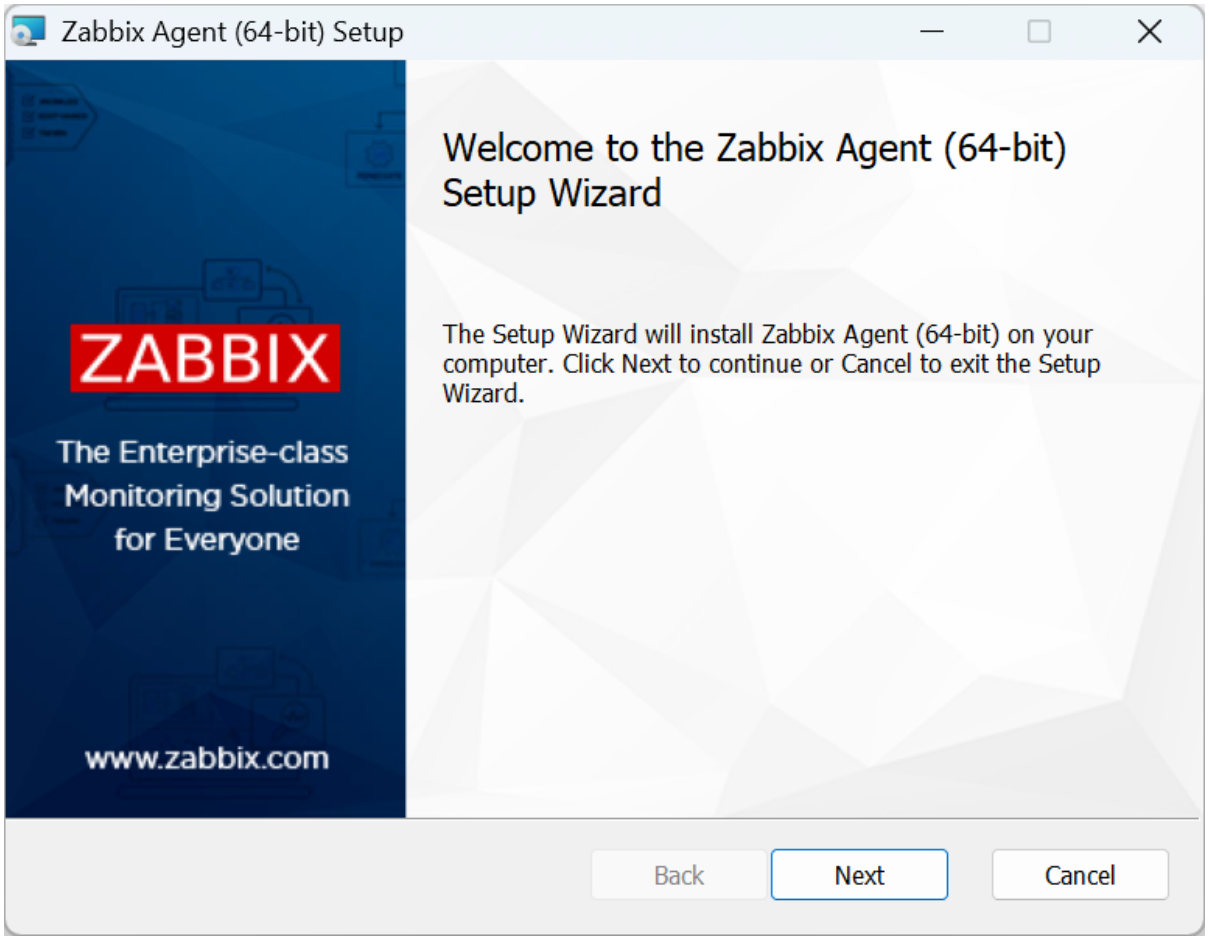
Chociaż instalacja przy użyciu pakietów MSI jest w pełni obsługiwana, zaleca się zainstalowanie co najmniej Microsoft .NET Framework 2, aby zapewnić prawidłową obsługę błędów.

Attention:
Zaleca się użycie domyślnej lokalizacji instalacji oferowanej przez instalator. Użycie niestandardowej lokalizacji bez wymaganych uprawnień może obniżyć bezpieczeństwo instalacji.

Instalacja za pomocą kreatora konfiguracji

Poniższe kroki instalacji dotyczą zarówno Zabbix agent, jak i Zabbix agent 2.

1. Kliknij dwukrotnie pobrany plik MSI, aby rozpocząć instalację:

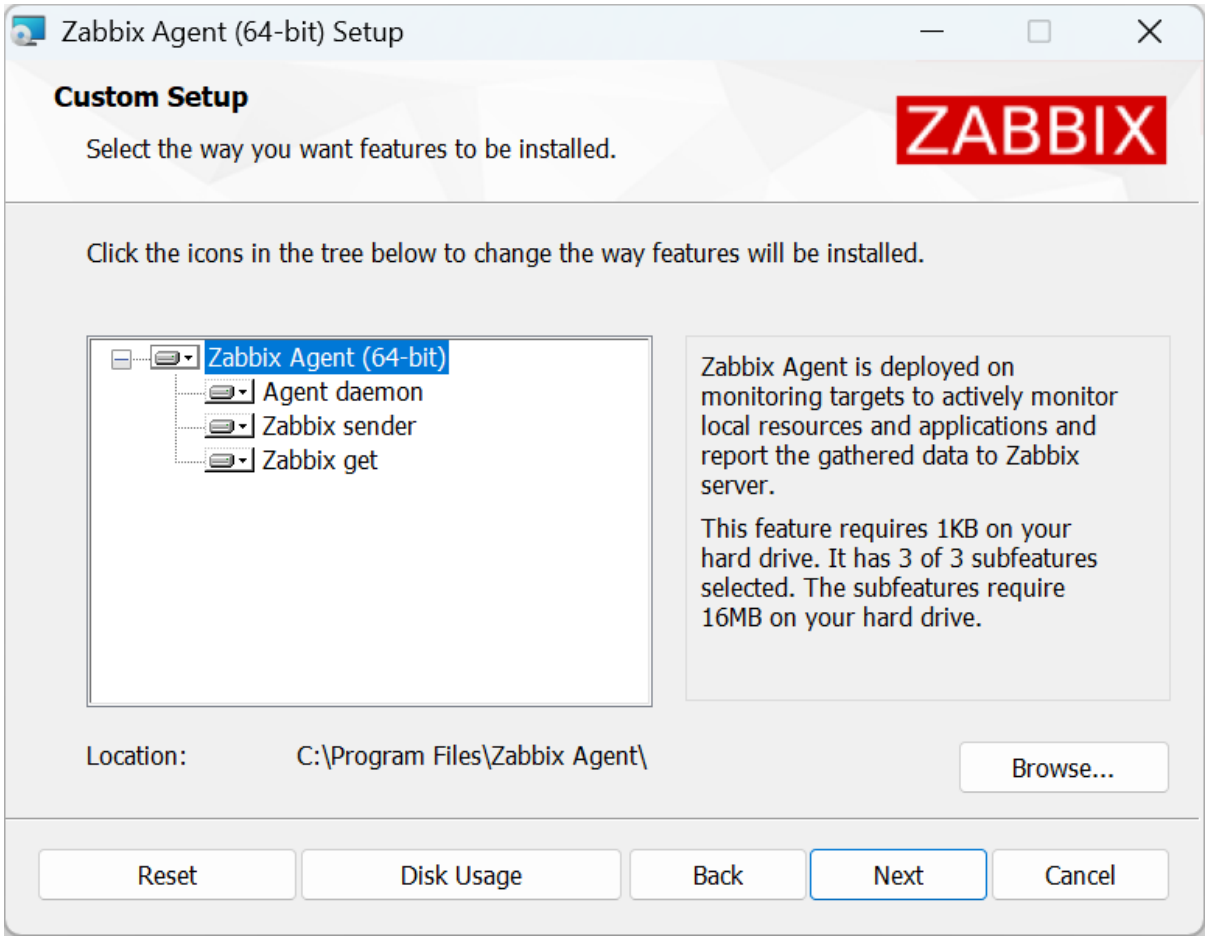


2. Zaakceptuj umowę licencyjną użytkownika końcowego:



3. Wybierz komponenty Zabbix (agent daemon, Zabbix sender, Zabbix get), które mają zostać zainstalowane:

Attention:
Zaleca się użycie domyślnej lokalizacji instalacji podanej przez instalator. Użycie niestandardowej lokalizacji bez wymaganych uprawnień może zagrozić bezpieczeństwu instalacji.

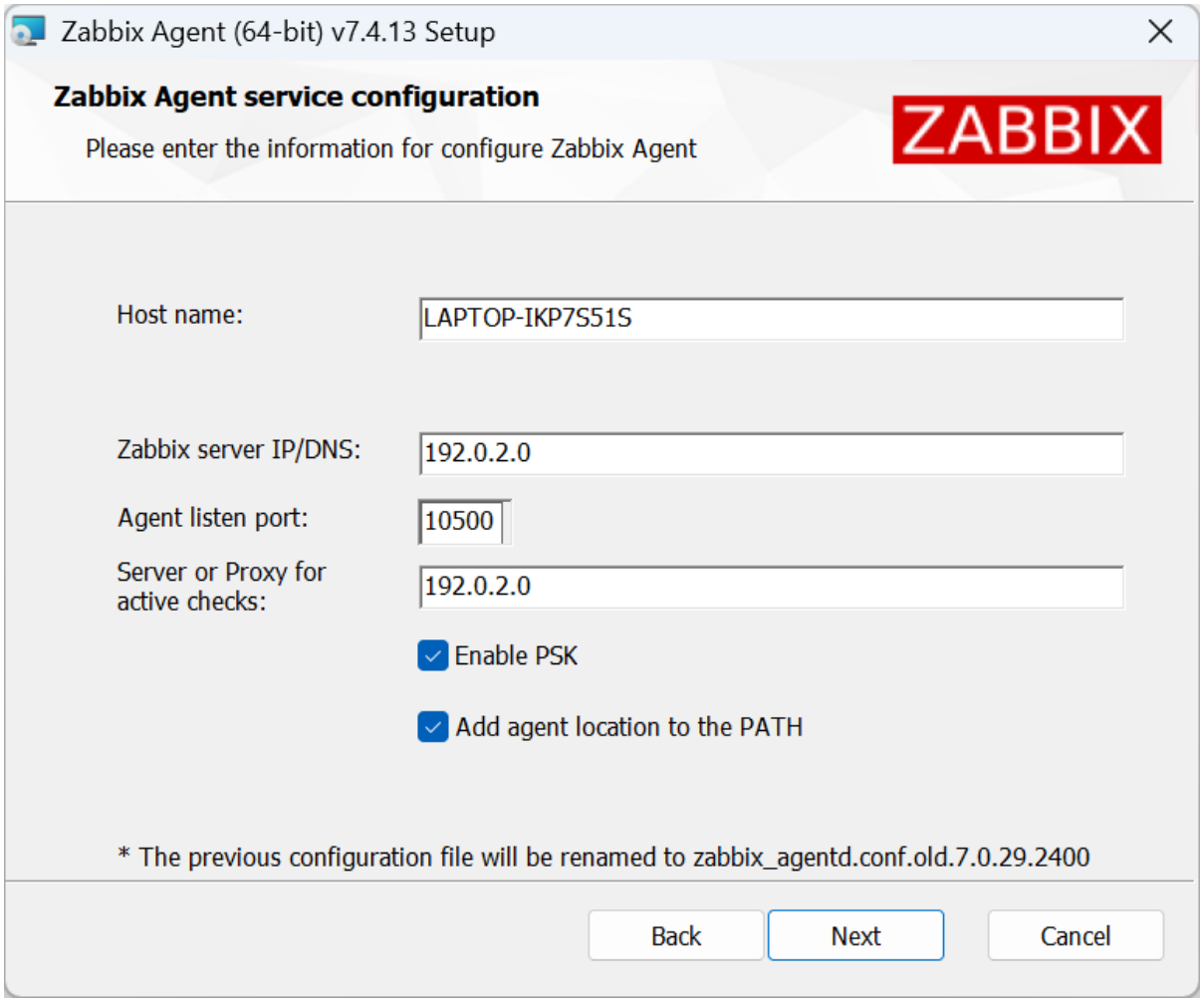


Note:
Jeśli używana jest niestandardowa lokalizacja poza katalogiem `Program Files`, a folder docelowy już istnieje, instalator nie zmieni jego uprawnień. Podczas nowej instalacji zostanie wyświetlony monit o ręczne ustawienie odpowiednich uprawnień lub kontynuowanie bez ich modyfikowania.

4. Skonfiguruj następujące parametry. Ich wartości zostaną ustawione w pliku konfiguracyjnym agent Zabbix:

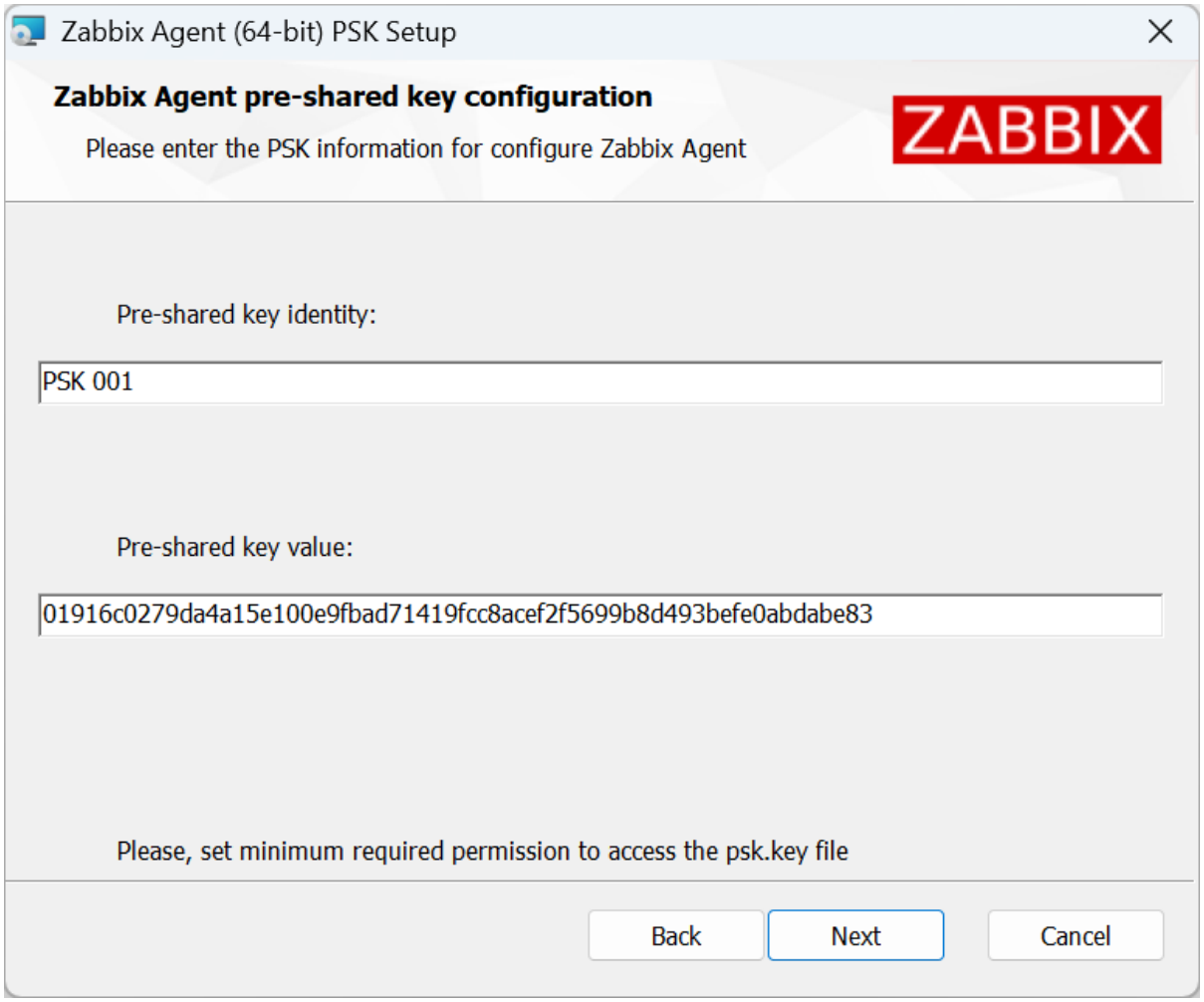
Parameter	Description
Host name	Nazwa hosta komputera, na którym instalowany jest Zabbix agent. Ustawia parametr Hostname.
Zabbix server IP/DNS	Lista rozdzielanych przecinkami adresów IP, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Ten parametr jest **obowiązkowy**. Ustawia parametr Server.
Agent listen port	agent będzie nasłuchiwać na tym porcie połączeń z serwera. Ustawia parametr ListenPort.
Server or Proxy for active checks	Adres serwera/proxy Zabbix lub konfiguracja klastra, z której mają być pobierane aktywne sprawdzenia. Adres serwera/proxy to adres IP lub nazwa DNS oraz opcjonalny port oddzielony dwukropkiem. Ustawia parametr ServerActive.
Enable PSK	Zaznacz pole wyboru, aby włączyć obsługę TLS z użyciem kluczy wstępnie współdzielonych. Ustawia parametry TLSConnect i TLSAccept na `psk`.
Add agent location to the PATH	Zaznacz pole wyboru, aby dodać lokalizację Zabbix agent do systemowej zmiennej PATH.

Note:
Jeśli zostanie wykryty istniejący Zabbix agent, zostaną wyświetlone wartości parametrów z jego pliku konfiguracyjnego. Ponadto istniejący plik konfiguracyjny zostanie przemianowany podczas instalacji, a następnie zostanie utworzony nowy plik konfiguracyjny.



5. Skonfiguruj parametry PSK, jeśli w poprzednim kroku zaznaczono pole wyboru *Enable PSK*. Parametry te również zostaną ustawione w pliku konfiguracyjnym Zabbix agent:

Parameter	Description
Pre-shared key identity	Ciąg identyfikatora klucza wstępnie współdzielonego. Ustawia parametr TLSPSKIdentity.
Pre-shared key value	Wartość ciągu klucza wstępnie współdzielonego. Tworzy plik `psk.key` zawierający klucz i ustawia parametr TLSPSKFile na lokalizację klucza (domyślnie: `C:\Program Files\Zabbix Agent\psk.key`). **Zaleca się** ograniczenie dostępu do pliku klucza wstępnie współdzielonego przez dostosowanie ustawień zabezpieczeń pliku tak, aby tylko Zabbix agent (lub użytkownik uruchamiający agent) mógł go odczytać.

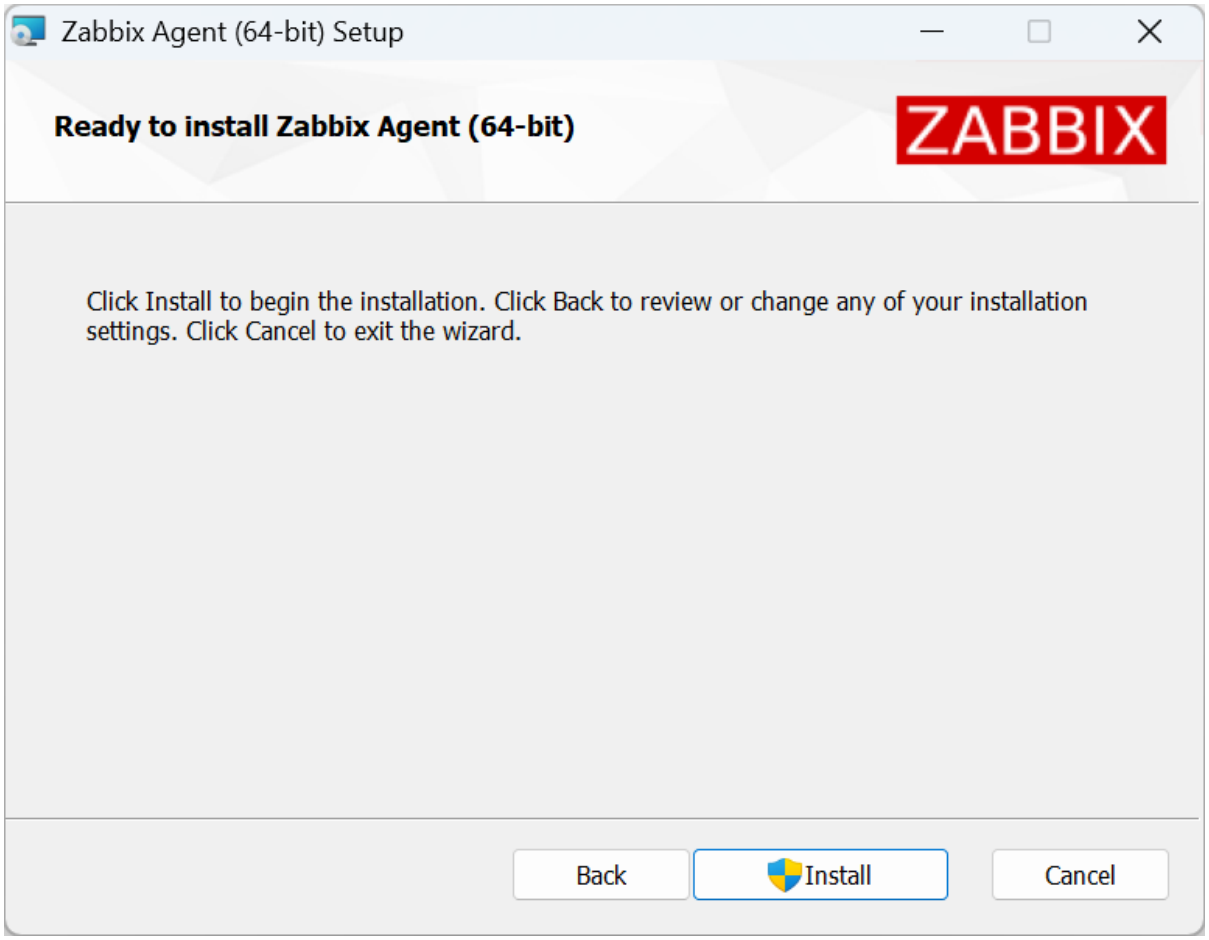


6. Kliknij przycisk *Install*, aby rozpocząć instalację.

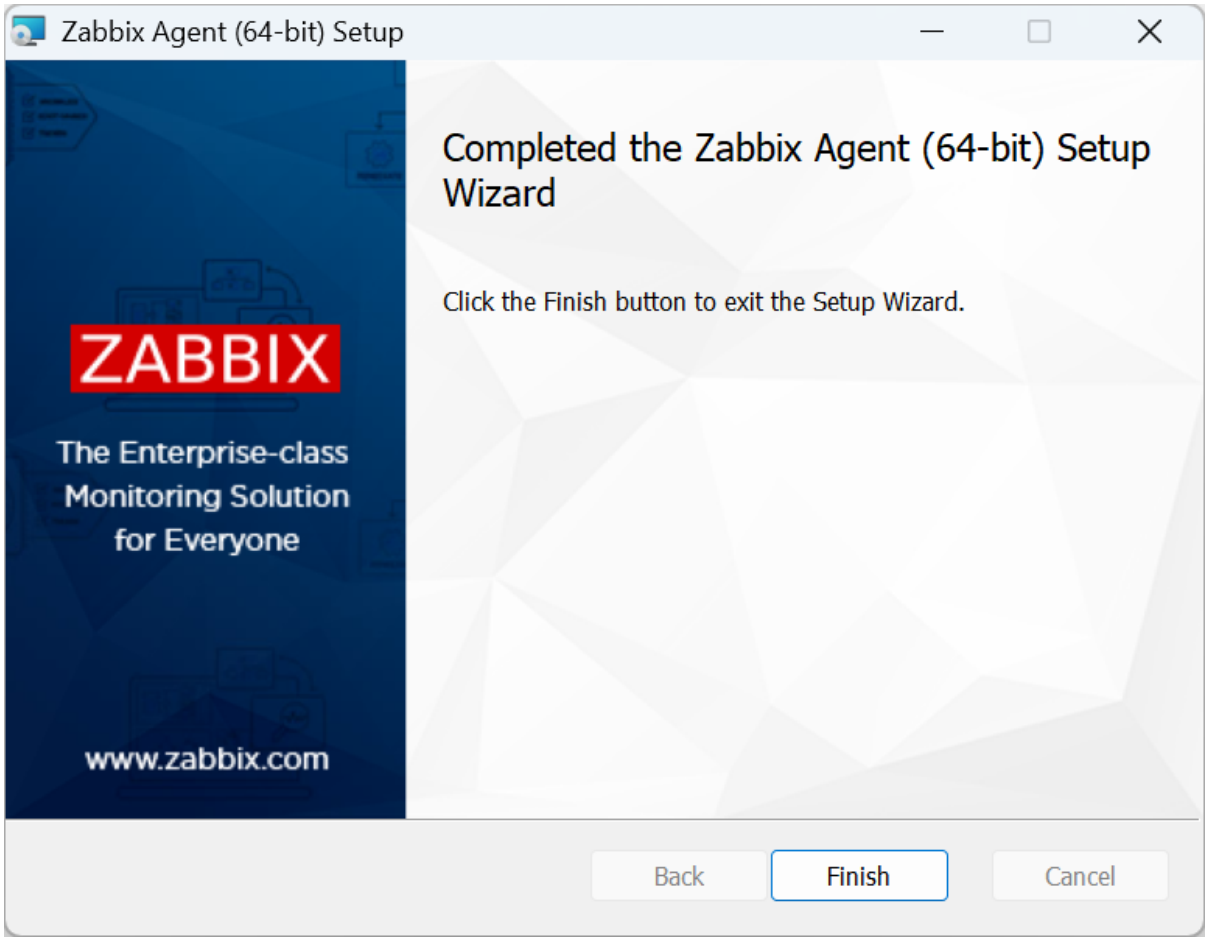
Wszystkie wybrane komponenty Zabbix oraz plik konfiguracyjny Zabbix agent zostaną zainstalowane w określonej lokalizacji (domyślnie: `C:\Program Files\Zabbix Agent`). To samo dotyczy Zabbix agent 2, z wyjątkiem dodatkowych plików konfiguracyjnych jego wbudowanych wtyczek, które zostaną zainstalowane w podfolderze `zabbix_agent2.d\plugins.d`.

Ponadto `zabbix_agentd.exe` (lub `zabbix_agent2.exe`) zostanie skonfigurowany jako usługa systemu Windows z opóźnionym automatycznym uruchamianiem (lub automatycznym uruchamianiem w wersjach systemu Windows starszych niż Windows Vista/Server 2008).

Jeśli podczas instalacji działa inna wersja Zabbix agent, zostanie wyświetlony monit o wybranie jednej z opcji: zamknięcie aplikacji i próba jej ponownego uruchomienia albo pozostawienie jej otwartej; w drugim przypadku wymagane będzie ponowne uruchomienie systemu.



7. Kliknij przycisk *Finish*, aby zamknąć kreatora konfiguracji.



Instalacja z wiersza poleceń

Agent Zabbix można zainstalować z wiersza poleceń, uruchamiając instalator MSI za pomocą msiexec. Na przykład:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.0-windows-amd64-openssl.msi" /qn+ SERVER=192.0.2
```

Ta metoda umożliwia instalacje bez nadzoru oraz niestandardowe konfiguracje z użyciem parametrów.

Obsługiwane parametry

Pakiety instalatora MSI Zabbix agent obsługują następujące parametry zarówno dla Zabbix agent, jak i Zabbix agent 2.

Note:
Parametry Zabbix agent/agent2 są ustawiane w pliku konfiguracyjnym podczas instalacji. Kliknij nazwę parametru, aby wyświetlić jego szczegółowy opis i przykłady konfiguracji na stronie Zabbix agent (Windows). Dla Zabbix agent 2 zobacz stronę Zabbix agent 2 (Windows).

Parameter	Description
ADDDEFAULT	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane w domyślnej konfiguracji. Więcej informacji można znaleźć w właściwości ADDDEFAULT. Możliwe wartości: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL` Przykład: `ADDDEFAULT=AgentProgram,GetProgram`
ADDLOCAL	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane lokalnie. Więcej informacji można znaleźć w właściwości ADDLOCAL. Możliwe wartości: `AgentProgram`, `GetProgram`, `SenderProgram`, `ALL` Przykład: `ADDLOCAL=AgentProgram,SenderProgram`
ALLOWDENYKEY	Lista parametrów AllowKey lub DenyKey rozdzielonych średnikami, służąca do ograniczania sprawdzeń Zabbix agent. W razie potrzeby użyj ukośnika odwrotnego do poprzedzenia separatora (`\;`). Ustawia parametry AllowKey i DenyKey w pliku konfiguracyjnym agenta. Przykład: `ALLOWDENYKEY="AllowKey=system.run[more C:\Windows\System32\drivers\etc\hosts\\; echo 'File read complete'];DenyKey=system.run[*]"`
CONF	Pełna ścieżka do pliku konfiguracyjnego szablonu dla Zabbix agent. Podczas instalacji ten plik stanie się plikiem konfiguracyjnym agenta. Plik musi zawierać co najmniej parametry Server i LogFile. Przykład: `CONF="C:\full\path\to\example.conf"`
DONOTSTART	Użyj `DONOTSTART=1`, aby instalator MSI nie uruchamiał usługi Zabbix agent.
ENABLEPATH	Użyj `ENABLEPATH=1`, aby dodać lokalizację Zabbix agent do systemowej zmiennej PATH.
ENABLEPERSISTENTBUFFER	Tylko Zabbix agent 2. Włącza użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji.
HOSTINTERFACE	Opcjonalny parametr definiujący interfejs hosta.
HOSTMETADATA	Opcjonalny parametr definiujący metadane hosta.
HOSTMETADATAITEM	Opcjonalny parametr definiujący pozycję używaną do pobierania metadanych hosta.
HOSTNAME	Opcjonalny parametr definiujący nazwę hosta.
HOSTNAMEITEM	Opcjonalny parametr definiujący pozycję używaną do pobierania nazwy hosta.
INCLUDE	Lista pojedynczych plików lub wszystkich plików w katalogu, rozdzielonych średnikami, które mają zostać dołączone do pliku konfiguracyjnego Zabbix agent.
IGNOREALREADYEXISTSFOLDER	W przypadku nowych instalacji użyj `IGNOREALREADYEXISTSFOLDER=1`, aby zezwolić na instalację w istniejącym folderze docelowym poza katalogiem `Program Files`. Jeśli ten parametr nie zostanie określony, instalacja zakończy się niepowodzeniem, gdy taki folder docelowy już istnieje.
INSTALLFOLDER	Pełna ścieżka do folderu, w którym zostaną zainstalowane komponenty Zabbix oraz plik konfiguracyjny Zabbix agent. W przypadku Zabbix agent 2 dodatkowe pliki konfiguracyjne dla wbudowanych wtyczek zostaną zainstalowane w podfolderze `zabbix_agent2.d\plugins.d`. Przykład: `INSTALLFOLDER="C:\Program Files\Zabbix Agent"`
LISTENIP	Lista adresów IP rozdzielonych przecinkami, na których agent ma nasłuchiwać.
LISTENPORT	Agent będzie nasłuchiwał na tym porcie połączeń z serwera.
LOGFILE	Nazwa pliku dziennika Zabbix agent.
LOGTYPE	Typ wyjścia dziennika.
NONMSICONFNAME	Pełna ścieżka do niestandardowego pliku konfiguracyjnego dla Zabbix agent. Podczas instalacji wszystkie prawidłowe parametry konfiguracji agenta obecne w tym pliku (ograniczone do tych wymienionych w tej tabeli) zostaną zapisane w nowo utworzonym pliku konfiguracyjnym agenta. Plik musi zawierać co najmniej parametr Server. Przykład: `NONMSICONFNAME="C:\full\path\to\example.conf"`
PERSISTENTBUFFERFILE	Tylko Zabbix agent 2. Plik, w którym Zabbix agent 2 ma przechowywać bazę danych SQLite.
PERSISTENTBUFFERPERIOD	Tylko Zabbix agent 2. Okres czasu, przez który dane mają być przechowywane, gdy nie ma połączenia z serwerem lub proxy.

Parameter	Description
SERVER	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix albo proxy Zabbix. Ten parametr jest **obowiązkowy**, z wyjątkiem sytuacji, gdy STARTAGENTS ma wartość 0.
SERVERACTIVE	Adres serwera Zabbix/proxy lub konfiguracja klastra, z którego mają być pobierane aktywne sprawdzenia.
SKIP	Użyj `SKIP=fw`, aby instalator MSI nie dodawał reguły wyjątku Zapory systemu Windows dla Zabbix agent.
SOURCEIP	Adres IP źródłowy dla połączeń wychodzących do serwera Zabbix lub proxy Zabbix, albo do nawiązywania połączeń podczas wykonywania niektórych pozycji (web.page.get, net.tcp.port itp.).
STARTAGENTS	Liczba wstępnie rozwidlonych instancji zabbix_agentd, które obsługują pasywne sprawdzenia. Jeśli ustawiono wartość 0, pasywne sprawdzenia są wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP.
STARTUPTYPE	Typ uruchamiania usługi Zabbix agent. Możliwe wartości: **automatic** - uruchamiaj usługę automatycznie podczas startu systemu Windows; **delayed** - *(domyślnie)* opóźnij uruchomienie usługi do czasu zakończenia startu usług uruchamianych automatycznie (dostępne w systemach Windows Vista/Server 2008 i nowszych wersjach); **manual** - uruchamiaj usługę ręcznie (przez użytkownika lub aplikację); **disabled** - wyłącz usługę, aby nie mogła zostać uruchomiona przez użytkownika lub aplikację. Przykład: `STARTUPTYPE=disabled`
STATUSPORT	Tylko Zabbix agent 2. Jeśli ustawiono, agent będzie nasłuchiwał na tym porcie na żądania stanu HTTP (http://localhost:<port>/status).
TIMEOUT	Określa, jak długo czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy Zabbix lub serwerem.
TLSACCEPT	Połączenia przychodzące, które mają być akceptowane (używane dla pasywnych sprawdzeń). Jeśli ustawiono `psk`, to TLSCONNECT również zostanie ustawiony na `psk` (o ile nie określono inaczej).
TLSCAFILE	Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer.
TLSCERTFILE	Pełna ścieżka do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów.
TLSCONNECT	Sposób, w jaki agent ma łączyć się z serwerem Zabbix lub proxy (używane dla aktywnych sprawdzeń). Jeśli ustawiono `psk`, to TLSACCEPT również zostanie ustawiony na `psk` (o ile nie określono inaczej).
TLSCRLFILE	Pełna ścieżka do pliku zawierającego certyfikaty unieważnione.
TLSKEYFILE	Pełna ścieżka do pliku zawierającego prywatny klucz Zabbix agent.
TLSPSKFILE	Pełna ścieżka do pliku zawierającego klucz wstępnie współdzielony Zabbix agent. Jeśli ustawiono zarówno TLSPSKFILE, jak i TLSPSKVALUE, wartość TLSPSKVALUE zostanie zapisana do pliku określonego w TLSPSKFILE. **Zaleca się** ograniczenie dostępu do pliku klucza wstępnie współdzielonego poprzez dostosowanie ustawień zabezpieczeń pliku tak, aby mógł go odczytać tylko Zabbix agent (lub użytkownik uruchamiający agenta).
TLSPSKIDENTITY	Łańcuch identyfikatora klucza wstępnie współdzielonego.
TLSPSKVALUE	Wartość klucza wstępnie współdzielonego. Jeśli ustawiono zarówno TLSPSKFILE, jak i TLSPSKVALUE, wartość TLSPSKVALUE zostanie zapisana do pliku określonego w TLSPSKFILE. Przykład: `TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952`
TLSSERVERCERTISSUER	Dozwolony wystawca certyfikatu serwera (proxy).
TLSSERVERCERTSUBJECT	Dozwolony podmiot certyfikatu serwera (proxy).
UNSAFEUSERPARAMETERS	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.

Przykłady

Poniższy przykład instaluje agent Zabbix z niestandardową konfiguracją. Włącza również obsługę TLS z użyciem kluczy współdzielonych.

```
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[*]"
```

Następny przykład instaluje nowszą wersję agenta Zabbix i używa **pliku konfiguracyjnego szablonu** (`CONF="C:\agent-template.conf"`). Podczas instalacji ten plik stanie się plikiem konfiguracyjnym agenta. Aby odziedziczyć parametry ze starego pliku konfiguracyjnego, użyj symboli zastępczych parametrów (np. `[AllowDenyKey]`).

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

#### przykład agent-template.conf:
LogFile=[LogFile]
[AllowDenyKey]
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
[Include]
[TLSConnect]
[TLSAccept]
[TLSPSKIdentity]
[TLSPSKFile]
```

Alternatywnie można użyć **własnego pliku konfiguracyjnego** (`NONMSICONFNAME="C:\agent-custom.conf"`). Podczas instalacji wszystkie prawidłowe parametry konfiguracji agenta obecne w tym pliku (ograniczone do tych wymienionych w powyższej tabeli) zostaną zapisane w nowo utworzonym pliku konfiguracyjnym agenta. Aby zachować istniejącą konfigurację agenta, zdefiniuj parametry, które mają zostać zachowane.

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent-8.0.1-windows-amd64-openssl.msi" /qn+ NONMSICONFNAME

#### przykład agent-custom.conf:
Server=192.0.2.8
Hostname=DESKTOP-X9F4A2B
```

Ładowalne wtyczki Zabbix agent 2

Ładowalne wtyczki Zabbix agent 2 można instalować w systemie Windows za pomocą 64-bitowych pakietów instalatora MSI, dostępnych do pobrania.

Minimalne wymagania systemowe dla instalacji MSI to Windows 10 (64-bit) lub Windows Server 2016.

> **Attention:**
> Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać szczegółowe wymagania oraz instrukcje instalacji.

Podobnie jak w przypadku Zabbix agent/agent2, ładowalne wtyczki można instalować za pomocą kreatora instalacji lub z wiersza poleceń.

Instalacja za pomocą kreatora konfiguracji

1. Kliknij dwukrotnie pobrany plik MSI, aby rozpocząć instalację.
2. Zaakceptuj umowę licencyjną użytkownika końcowego.
3. Wybierz ładowalne wtyczki Zabbix agent 2 do zainstalowania.

> **Attention:**
> Zaleca się użycie domyślnej lokalizacji instalacji udostępnionej przez instalator. Użycie niestandardowej lokalizacji bez wymaganych uprawnień może zagrozić bezpieczeństwu instalacji.

4. Kliknij *Install*, aby rozpocząć instalację.

Wszystkie wybrane ładowalne wtyczki Zabbix agent 2 zostaną zainstalowane w określonej lokalizacji (domyślnie: `C:\Program Files\Zabbix Agent 2`), a ich pliki konfiguracyjne zostaną zainstalowane w podfolderze `zabbix_agent2.d`.

5. Kliknij przycisk *Finish*, aby zamknąć kreatora konfiguracji.

Instalacja z wiersza poleceń

Wtyczki ładowalne Zabbix agent 2 można zainstalować z wiersza poleceń, uruchamiając instalator MSI za pomocą `msiexec`. Na przykład:

```
msiexec.exe /l*v "C:\package.log" /i "C:\zabbix_agent2_plugins-8.0.0-windows-amd64.msi" /qn+
```

Pakiety instalatora MSI dla ładowalnych wtyczek Zabbix agent 2 obsługują następujące parametry.

Parameter	Description
ADDDEFAULT	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane w domyślnej konfiguracji. Więcej informacji można znaleźć w właściwości ADDDEFAULT. Możliwe wartości: `ALL`, `CephPlugin`, `EmberplusPlugin`, `MongodbPlugin`, `MssqlPlugin`, `NvidiagpuPlugin`, `PostgresqlPlugin` Przykład: `ADDDEFAULT=MongodbPlugin,PostgresqlPlugin`
ADDLOCAL	Lista komponentów rozdzielonych przecinkami, które mają zostać zainstalowane lokalnie. Więcej informacji można znaleźć w właściwości ADDLOCAL. Możliwe wartości: `ALL`, `CephPlugin`, `EmberplusPlugin`, `MongodbPlugin`, `MssqlPlugin`, `NvidiagpuPlugin`, `PostgresqlPlugin` Przykład: `ADDLOCAL=MongodbPlugin,MssqlPlugin`
INSTALLFOLDER	Pełna ścieżka do folderu, w którym zostaną zainstalowane komponenty Zabbix, a ich pliki konfiguracyjne zostaną umieszczone w podfolderze `zabbix_agent2.d`. Przykład: `INSTALLFOLDER="C:\Program Files\Zabbix Agent 2"`

Budowanie Zabbix agent 2 w systemie Windows

Omówienie

Ta strona pokazuje, jak zbudować agent Zabbix 2 ze źródeł w systemie Windows 10 (64-bitowym lub 32-bitowym).

Zarówno wersję 32-bitową, jak i 64-bitową można zbudować na platformie 64-bitowej, ale na platformie 32-bitowej można zbudować tylko wersję 32-bitową.

Do zbudowania agenta Zabbix 2 wymagane są:

- narzędzia kompilacyjne MinGW
- język programowania Go
- OpenSSL (dla funkcji szyfrowania w Zabbix)
- PCRE2 (Perl Compatible Regular Expressions; dla funkcji dopasowywania wzorców wyrażeń regularnych w Zabbix)

Agenta Zabbix 2 można zbudować przy użyciu jednej z następujących metod:

- Użycie vcpkg - zautomatyzowane podejście upraszczające zarządzanie zależnościami za pomocą menedżera pakietów C++.
- Kompilacja ręczna - ręczne podejście wymagające zainstalowania wszystkich zależności przed skompilowaniem agenta.

Attention:
Przed rozpoczęciem procesu kompilacji należy pamiętać:

- Do wykonywania poleceń używaj wiersza polecenia uruchomionego przez użytkownika z wystarczającymi uprawnieniami do zapisu w chronionych folderach. Jednak podczas instalowania OpenSSL i PCRE2 używaj terminala MSYS2.
- Zaleca się utworzenie katalogu roboczego `C:\Zabbix` na wszystkie pliki źródłowe i foldery kompilacji. Jednak skompilowane komponenty należy zainstalować w `C:\Zabbix\x64` (lub `C:\Zabbix\x86` w przypadku kompilacji 32-bitowych).

Budowanie agenta Zabbix 2 za pomocą vcpkg

Ta sekcja zawiera instrukcje dotyczące budowania agenta Zabbix za pomocą vcpkg, menedżera pakietów upraszczającego zarządzanie zależnościami i integrację z projektami C++.

1. Pobierz i zainstaluj Build Tools for Visual Studio 2022. Podczas instalacji wybierz obciążenie *Desktop development with C++*, które obejmuje menedżera pakietów vcpkg.

2. Pobierz i zainstaluj Go (dostępny jako instalator MSI). Podczas instalacji określ `C:\Zabbix\Go` jako folder instalacyjny.

3. Pobierz dystrybucję MinGW korzystającą z biblioteki wykonawczej Microsoft Visual C; na przykład:

- Dla kompilacji 64-bitowych: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Dla kompilacji 32-bitowych: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

Attention:
Wersja MinGW musi spełniać minimalne wymagania.

Następnie wypakuj ją do `C:\Zabbix\mingw64` (lub `C:\Zabbix\mingw32` dla kompilacji 32-bitowych).

4. Zainicjuj vcpkg i zainstaluj zależności wymagane do zbudowania agenta Zabbix 2 (pamiętaj, że może to potrwać):

```
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

#### Dla kompilacji 64-bitowych:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
vcpkg install --triplet x64-mingw-static --x-install-root=x64

#### Dla kompilacji 32-bitowych:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
vcpkg install --triplet x86-mingw-static --x-install-root=x86
```

5. Pobierz archiwum źródłowe Zabbix i wypakuj je do `C:\Zabbix\zabbix-8.0.0`.

6. Przejdź do katalogu kompilacji Zabbix (`C:\Zabbix\zabbix-8.0.0\build\mingw`) i utwórz następujący skrypt `build.bat`:

- Dla kompilacji 64-bitowych:

```
:: Add MinGW and Go to the system `PATH` variable for the current session:
set PATH=C:\Zabbix\mingw64\bin;%PATH%
set PATH=C:\Zabbix\Go\bin;%PATH%

:: Set vcpkg installation path:
set vcpkg="C:\Zabbix\x64\x64-mingw-static"

:: Set linker flags for Crypt32 library:
SET CGO_LDFLAGS="-lCrypt32"

:: Run the build process:
mingw32-make GOFLAGS="-buildvcs=false" ARCH=AMD64 ^
    PCRE2="%vcpkg%" ^
    OPENSSL="%vcpkg%" ^
    all
```

- Dla kompilacji 32-bitowych:

```
:: Add MinGW and Go to the system `PATH` variable for the current session:
set PATH=C:\Zabbix\mingw32\bin;%PATH%
set PATH=C:\Zabbix\Go\bin;%PATH%

:: Set vcpkg installation path:
set vcpkg="C:\Zabbix\x86\x86-mingw-static"

:: Set linker flags for Crypt32 library:
SET CGO_LDFLAGS="-lCrypt32"

:: Run the build process:
mingw32-make GOFLAGS="-buildvcs=false" ARCH=x86 ^
    PCRE2="%vcpkg%" ^
    OPENSSL="%vcpkg%" ^
    all
```

7. Skompiluj agenta Zabbix 2, wykonując skrypt:

```
build.bat
```

Po kompilacji plik binarny agenta Zabbix 2 będzie znajdować się w `C:\Zabbix\zabbix-8.0.0\bin\win64` (dla kompilacji 64-bitowych) lub `C:\Zabbix\zabbix-8.0.0\bin\win32` (dla kompilacji 32-bitowych). Pliki konfiguracyjne agenta Zabbix 2 znajdują się w `C:\Zabbix\zabbix-8.0.0\src\go\conf`.

Aby uruchomić agenta, skopiuj `zabbix_agent2.exe` i jego pliki konfiguracyjne do dedykowanego folderu (np. `C:\Zabbix\agent2`), a następnie uruchom agenta:

```
mkdir C:\Zabbix\agent2

#### Dla kompilacji 64-bitowych:
copy C:\Zabbix\zabbix-8.0.0\bin\win64\zabbix_agent2.exe C:\Zabbix\agent2\

#### Dla kompilacji 32-bitowych:
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
```

W razie potrzeby kontynuuj, kompilując ładowalne wtyczki agenta Zabbix 2.

Kompilowanie ładowalnych wtyczek Zabbix agent 2

1. Pobierz źródła wtyczki Zabbix zgodne z wersją Zabbix agent 2 (np. `zabbix-agent2-plugin-postgresql-8.0.0.tar.gz`) i rozpakuj je do `C:\Zabbix`.

> **Attention:**
> Przed zainstalowaniem wtyczki sprawdź plik README. Może on zawierać szczegółowe wymagania i instrukcje instalacji.

2. Przejdź do rozpakowanego katalogu wtyczki i skompiluj wtyczkę:

```
cd C:\Zabbix\zabbix-agent2-plugin-ember-plus-8.0.0

#### Dla kompilacji 64-bitowej:
mingw32-make ARCH=AMD64

#### Dla kompilacji 32-bitowej:
mingw32-make ARCH=x86
```

Po kompilacji plik binarny wtyczki Zabbix agent 2 oraz jej plik konfiguracyjny będą znajdować się w tym samym katalogu wtyczki.

Plik wykonywalny wtyczki może znajdować się w dowolnym miejscu, o ile może być załadowany przez Zabbix agent 2. Określ ścieżkę do pliku binarnego wtyczki w pliku konfiguracyjnym wtyczki, np. w pliku postgresql.conf dla wtyczki PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

Ścieżka do pliku konfiguracyjnego wtyczki musi być określona w parametrze Include pliku konfiguracyjnego Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Przejdź do konfiguracji, aby uzyskać więcej informacji na temat konfigurowania wtyczek.

Ręczne budowanie Zabbix agent 2

> **Attention:**
> Ta metoda budowania Zabbix agent 2 jest odpowiednia dla użytkowników, którzy wymagają pełnej kontroli nad środowiskiem budowania lub działają w ograniczonym środowisku, w którym użycie vcpkg nie jest możliwe.

Ta sekcja zawiera instrukcje dotyczące ręcznego budowania Zabbix agent 2, co obejmuje instalację wymaganych narzędzi do budowania i zależności, a następnie kompilację agenta.

Konfiguracja narzędzi do budowania

1. Pobierz i zainstaluj MSYS2 (dostępny jako instalator MSI). Podczas instalacji upewnij się, że jako folder instalacyjny wskazano `C:\Zabbix\msys64`.

2. Pobierz i zainstaluj Go (dostępny jako instalator MSI; zobacz aktualnie obsługiwane wersje Go). Podczas instalacji upewnij się, że jako folder instalacyjny wskazano `C:\Zabbix\Go`.

3. Pobierz dystrybucję MinGW, która korzysta z biblioteki uruchomieniowej Microsoft Visual C; na przykład:

- Dla kompilacji 64-bitowych: `x86_64-15.1.0-release-win32-seh-msvcrt-rt_v12-rev0.7z`
- Dla kompilacji 32-bitowych: `i686-15.1.0-release-win32-dwarf-msvcrt-rt_v12-rev0.7z`

Attention:
Wersja MinGW musi spełniać minimalne wymagania.

Następnie rozpakuj ją do `C:\Zabbix\mingw64` (lub `C:\Zabbix\mingw32` dla kompilacji 32-bitowych).

Instalowanie OpenSSL

Note:
Aby skompilować agent Zabbix bez obsługi TLS, przejdź do sekcji Instalowanie PCRE2.

1. Otwórz terminal MSYS2 MSYS z uprawnieniami administratora i uruchom następujące polecenia:

```
pacman -S perl-Locale-Maketext-Simple
pacman -S nasm
pacman -S make
pacman -S cmake
```

2. Pobierz archiwum źródłowe OpenSSL i rozpakuj je do katalogu `C:\Zabbix\openssl-3.5.0`.

3. Przejdź do rozpakowanego katalogu OpenSSL i utwórz następujący skrypt `build.sh`:

- Dla kompilacji 64-bitowych:

```
####!/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
```

- Dla kompilacji 32-bitowych:

```
####!/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:
Upewnij się, że użytkownikom niebędącym administratorami odebrano uprawnienia do zapisu w katalogu `C:\Zabbix\x64\OpenSSL-Win64-350-static`. W przeciwnym razie agent załaduje ustawienia SSL ze ścieżki, którą mogą modyfikować użytkownicy bez uprawnień administratora, co może prowadzić do luki w zabezpieczeniach.

- Opcja `no-shared` sprawia, że biblioteki statyczne OpenSSL libcrypto.lib i libssl.lib są samowystarczalne, dzięki czemu pliki binarne Zabbix zawierają OpenSSL i nie wymagają zewnętrznych bibliotek DLL. Oznacza to, że pliki binarne Zabbix można kopiować na inne komputery z systemem Windows bez bibliotek OpenSSL; jednak po wydaniu nowej wersji OpenSSL zawierającej poprawki błędów agent Zabbix będzie wymagać ponownej kompilacji.
- Bez opcji `no-shared` Zabbix korzysta w czasie działania z bibliotek DLL OpenSSL. Oznacza to, że aktualizacje OpenSSL mogą nie wymagać ponownej kompilacji Zabbix; jednak podczas kopiowania go na inne komputery należy skopiować również biblioteki DLL OpenSSL.

Więcej informacji o innych opcjach konfiguracji OpenSSL można znaleźć w dokumentacji OpenSSL.

4. Skonfiguruj i zainstaluj OpenSSL, wykonując skrypt (pamiętaj, że może to potrwać pewien czas):

```
cd /c/Zabbix/openssl-3.5.0
./build.sh
```

Instalacja PCRE2

1. Pobierz archiwum źródłowe PCRE2 i rozpakuj je do `C:\Zabbix\pcre2-10.45`.

2. Otwórz terminal MSYS2 MSYS z uprawnieniami administratora. Następnie utwórz katalog `build` w rozpakowanym katalogu PCRE2 i przejdź do niego:

```
mkdir /c/Zabbix/pcre2-10.45/build
cd /c/Zabbix/pcre2-10.45/build
```

3. Skonfiguruj PCRE2:

```
#### Dla kompilacji 64-bitowych:
export PATH="/c/Zabbix/mingw64/bin:$PATH"
cmake -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-O2 -g" -DCMAKE_INSTALL_PREFIX="/c/Zabbix/x64/PCRE2" ..

#### Dla kompilacji 32-bitowych:
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:
Jeśli wystąpią jakiekolwiek błędy, zaleca się usunięcie pamięci podręcznej CMake przed ponowną próbą przeprowadzenia procesu budowania CMake. Pamięć podręczna (`CMakeCachecache.txt`) może znajdować się w katalogu build rozpakowanego katalogu PCRE2.

4. Zainstaluj PCRE2:

```
make install
```

Kompilowanie Zabbix agent 2

1. Pobierz archiwum źródłowe Zabbix i rozpakuj je do `C:\Zabbix\zabbix-8.0.0`.

Jeśli potrzebujesz wygenerować archiwum źródłowe z surowego repozytorium źródłowego (np. aby zastosować niestandardowe poprawki lub skompilować najnowszy kod źródłowy), uruchom następujące polecenia na komputerze **Linux** z zainstalowanym Go (wymaganym do konfiguracji Zabbix agent 2):

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure
make dist
```

Spowoduje to utworzenie archiwum źródłowego, które można następnie skopiować na komputer z systemem Windows.

2. Otwórz wiersz polecenia z uprawnieniami administratora. Następnie przejdź do katalogu kompilacji Zabbix i skompiluj Zabbix agent; upewnij się, że prawidłowo określono katalogi, w których zainstalowano OpenSSL i PCRE2:

- Kompilacje 64-bitowe:

```
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

#### Z obsługą TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2" OPENSSL="C:\Zabbix\x64\OpenSSL-Win64-350-static"

#### Bez obsługi TLS:
mingw32-make ARCH=AMD64 PCRE2="C:\Zabbix\x64\PCRE2"
```

- Kompilacje 32-bitowe:

```
cd C:\Zabbix\zabbix-8.0.0\build\mingw
set PATH=C:\Zabbix\mingw32\bin;%PATH%

#### Z obsługą TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2" OPENSSL="C:\Zabbix\x86\OpenSSL-Win32-350-static"
```

```
#### Bez obsługi TLS:
mingw32-make ARCH=x86 PCRE2="C:\Zabbix\x86\PCRE2"
```

Po kompilacji plik binarny Zabbix agent 2 będzie znajdować się w `C:\Zabbix\zabbix-8.0.0\bin\win64` (lub w `C:\Zabbix\zabbix-8.0.` w przypadku kompilacji 32-bitowych). Pliki konfiguracyjne Zabbix agent 2 znajdują się w `C:\Zabbix\zabbix-8.0.0\src\go\conf`.

Aby uruchomić agenta, skopiuj plik binarny `zabbix_agent2.exe` i jego pliki konfiguracyjne do dedykowanego folderu (np. `C:\Zabbix\agent2`), a następnie uruchom agenta:

```
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
```

W razie potrzeby przejdź do sekcji kompilowanie ładowalnych wtyczek Zabbix agent 2.

Building Zabbix agent on Windows

Przegląd

Ta strona pokazuje, jak zbudować agent Zabbix ze źródeł w systemie Windows 10 (64-bitowym).

Te instrukcje dotyczą wersji systemu Windows obsługujących Visual Studio 2022.

Do zbudowania agenta Zabbix wymagane są:

- Kompilator C (zawarty w Build Tools for Visual Studio 2022)
- OpenSSL (na potrzeby funkcji szyfrowania w Zabbix)
- PCRE2 (wyrażenia regularne zgodne z Perlem; na potrzeby funkcji dopasowywania wzorców wyrażeń regularnych w Zabbix)

Agenta Zabbix można zbudować przy użyciu jednej z następujących metod:

- Użycie vcpkg - zautomatyzowane podejście upraszczające zarządzanie zależnościami za pomocą menedżera pakietów C++.
- Kompilacja ręczna - ręczne podejście wymagające zainstalowania wszystkich zależności przed skompilowaniem agenta.

W zależności od potrzeb związanych z monitorowaniem mogą być wymagane dodatkowe biblioteki. Więcej informacji można znaleźć w sekcji Wymagania.

Attention:
Przed rozpoczęciem procesu kompilacji należy pamiętać:

- Do wykonywania poleceń należy używać narzędzia „x64 Native Tools Command Prompt” (zawartego w Build Tools for Visual Studio 2022), uruchomionego przez użytkownika z wystarczającymi uprawnieniami do zapisu w chronionych folderach.
- Zaleca się utworzenie katalogu roboczego `C:\Zabbix` na wszystkie pliki źródłowe i foldery kompilacji. Jednak skompilowane komponenty powinny zostać zainstalowane w `C:\Program Files\Zabbix\x64`.

Kompilowanie agenta Zabbix za pomocą vcpkg

Ta sekcja zawiera instrukcje dotyczące kompilowania agenta Zabbix za pomocą vcpkg, menedżera pakietów upraszczającego zarządzanie zależnościami i integrację z projektami C++.

1. Pobierz i zainstaluj Build Tools for Visual Studio 2022. Podczas instalacji wybierz obciążenie *Desktop development with C++*, które zawiera narzędzia wymagane do kompilowania agenta za pomocą vcpkg:

- Kompilator C (Microsoft Visual C++)
- Narzędzie wiersza poleceń NMake
- Menedżer pakietów vcpkg
- „x64 Native Tools Command Prompt”

2. Otwórz `cmd.exe` z uprawnieniami administratora, zainicjuj vcpkg i zainstaluj zależności wymagane do kompilowania agenta Zabbix (pamiętaj, że może to potrwać pewien czas):

```
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. Pobierz archiwum źródłowe Zabbix i wypakuj je do `C:\Zabbix\zabbix-8.0.0`.

4. Przejdź do katalogu kompilacji Zabbix (`C:\Zabbix\zabbix-8.0.0\build\win32\project`) i utwórz następujący skrypt `build.bat`; upewnij się, że katalogi, w których zainstalowano OpenSSL i PCRE2, zostały określone prawidłowo:

```
:: Ustaw ścieżkę instalacji vcpkg:
set vcpkg=C:\Program Files\Zabbix\x64\x64-windows-static

:: Uruchom proces kompilacji:
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. Otwórz „x64 Native Tools Command Prompt" i skompiluj agenta Zabbix, wykonując skrypt:

```
cd C:\Zabbix\zabbix-8.0.0\build\win32\project
build.bat
```

Jeśli otwarto inny wiersz poleceń, najpierw zainicjuj środowisko kompilacji 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
```

Ścieżki zawierające spacje należy ujmować w cudzysłowy.

Po kompilacji pliki binarne komponentów Zabbix będą znajdować się w `C:\Zabbix\zabbix-8.0.0\bin\win64`. Plik konfiguracyjny agenta Zabbix znajduje się w `C:\Zabbix\zabbix-8.0.0\conf`.

Aby uruchomić agenta, skopiuj `zabbix_agentd.exe` i jego plik konfiguracyjny do dedykowanego folderu (np. `C:\Zabbix\agent`), a następnie uruchom agenta:

```
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
```

Ręczne budowanie Zabbix agent

Attention:
Ta metoda budowania Zabbix agent jest odpowiednia dla użytkowników, którzy wymagają pełnej kontroli nad środowiskiem budowania lub znajdują się w środowisku z ograniczeniami, gdzie użycie vcpkg nie jest możliwe.

Ta sekcja zawiera instrukcje dotyczące ręcznego budowania Zabbix agent, co obejmuje instalację wymaganych narzędzi do budowania i zależności (Perl, OpenSSL, PCRE2), a następnie kompilację agent.

Instalowanie narzędzi do kompilowania

1. Pobierz i zainstaluj Build Tools for Visual Studio 2022. Podczas instalacji wybierz obciążenie *Desktop development with C++*, które zawiera narzędzia wymagane do ręcznego kompilowania agent:

- Kompilator C (Microsoft Visual C++)
- Narzędzie wiersza poleceń NMake
- „x64 Native Tools Command Prompt"

Instalowanie OpenSSL

Note:
Aby skompilować Zabbix agent bez obsługi TLS, przejdź do sekcji Instalowanie PCRE2.

1. Pobierz i zainstaluj Strawberry Perl (dostępny jako instalator MSI). Podczas instalacji określ `C:\Zabbix\Strawberry` jako folder instalacyjny.

2. Zainstaluj moduł Perl `Text::Template`:

```
cpanm Text::Template
```

3. Sprawdź, czy podczas instalacji Strawberry Perl został skompilowany Netwide Assembler (NASM; wymagany do kompilowania OpenSSL):

```
nasm -v
#### NASM version 2.16.01 compiled on May  3 2024
```

Jeśli NASM nie został skompilowany, zainstaluj go ręcznie. Więcej informacji znajdziesz w dokumentacji NASM.

4. Pobierz archiwum źródłowe OpenSSL i wypakuj je do `C:\Zabbix\openssl-3.5.0`.

5. Przejdź do wypakowanego katalogu i skonfiguruj OpenSSL, na przykład:

```
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:
Jeśli podczas kompilowania Zabbix agent w systemie Windows wybierzesz niestandardowy katalog dla OpenSSL (np. `C:\zabbix` lub `C:\openssl-64bit`), upewnij się, że użytkownicy niebędący administratorami nie mają uprawnień do zapisu w tym katalogu. W przeciwnym razie agent załaduje ustawienia SSL ze ścieżki, którą mogą modyfikować użytkownicy bez uprawnień administratora, co może prowadzić do powstania luki w zabezpieczeniach.

- Opcja `no-shared` sprawia, że biblioteki statyczne OpenSSL libcrypto.lib i libssl.lib są samowystarczalne, dzięki czemu pliki binarne Zabbix zawierają OpenSSL i nie wymagają zewnętrznych bibliotek DLL. Oznacza to, że pliki binarne Zabbix można kopiować na inne komputery z systemem Windows bez bibliotek OpenSSL; jednak po wydaniu nowej wersji OpenSSL zawierającej poprawki błędów konieczne będzie ponowne skompilowanie Zabbix agent.
- Bez opcji `no-shared` Zabbix korzysta w czasie działania z bibliotek DLL OpenSSL. Oznacza to, że aktualizacje OpenSSL mogą nie wymagać ponownego kompilowania Zabbix agent; jednak podczas kopiowania go na inne komputery należy skopiować również biblioteki DLL OpenSSL.

Więcej informacji o innych opcjach konfiguracji OpenSSL znajdziesz w dokumentacji OpenSSL.

6. Skompiluj OpenSSL i uruchom testy (pamiętaj, że może to potrwać pewien czas):

Attention:
Uruchom testy bez uprawnień administratora; w przeciwnym razie może to prowadzić do nieoczekiwanych wyników lub zagrożeń bezpieczeństwa. Jeśli niektóre testy zakończą się niepowodzeniem, zapoznaj się z dokumentacją OpenSSL, aby znaleźć rozwiązanie problemu.

```
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. Zainstaluj OpenSSL:

```
nmake install
```

Aby zainstalować tylko składniki oprogramowania (biblioteki i pliki nagłówkowe, ale bez dokumentacji), możesz użyć polecenia `nmake install_sw`.

Instalacja PCRE2

1. Pobierz i zainstaluj CMake (dostępny jako instalator MSI). Podczas instalacji upewnij się, że jako folder instalacyjny wskazano `C:\Zabbix\CMake`, oraz wybierz opcję *Add CMake to the PATH environment variable*.

2. Pobierz archiwum kodu źródłowego PCRE2 i rozpakuj je do `C:\Zabbix\pcre2-10.45`.

3. Utwórz katalog `build` w rozpakowanym katalogu PCRE2 i przejdź do niego:

```
mkdir C:\Zabbix\pcre2-10.45\build
cd C:\Zabbix\pcre2-10.45\build
```

4. Skonfiguruj PCRE2:

```
cmake -G "NMake Makefiles" -DPCRE_SUPPORT_UNICODE_PROPERTIES=ON -DCMAKE_BUILD_TYPE=Release -DCMAKE_INSTALL
```

Note:
Jeśli wystąpią jakiekolwiek błędy, zaleca się usunięcie pamięci podręcznej CMake przed ponowną próbą przeprowadzenia procesu budowania CMake. Pamięć podręczna (`CMakeCachecache.txt`) znajduje się w katalogu build rozpakowanego katalogu PCRE2.

5. Zbuduj PCRE2 za pomocą NMake:

```
nmake
```

6. Zainstaluj PCRE2:

```
cmake --install .
```

Kompilowanie Zabbix agent

1. Pobierz archiwum źródłowe Zabbix i rozpakuj je do `C:\Zabbix\zabbix-8.0.0`.

Jeśli musisz wygenerować archiwum źródłowe z surowego repozytorium źródłowego (np. aby zastosować niestandardowe poprawki lub skompilować najnowszy kod źródłowy), uruchom następujące polecenia na komputerze **Linux**:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
cd zabbix
./bootstrap.sh
./configure
make dist
```

Spowoduje to utworzenie archiwum źródłowego, które można następnie skopiować na komputer z systemem Windows.

2. Przejdź do katalogu kompilacji Zabbix, otwórz „x64 Native Tools Command Prompt” i skompiluj Zabbix agent (lub inne komponenty); upewnij się, że prawidłowo określono katalogi, w których zainstalowano OpenSSL i PCRE2:

```
"%ProgramFiles(x86)%\Microsoft Visual Studio\2022\BuildTools\VC\Auxiliary\Build\vcvarsall.bat" x64

cd C:\Zabbix\zabbix-8.0.0\build\win32\project

#### Z obsługą 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

#### Bez obsługi 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
```

Ścieżki zawierające spacje należy ująć w cudzysłowy.

Po kompilacji pliki binarne komponentów Zabbix będą znajdować się w `C:\Zabbix\zabbix-8.0.0\bin\win64`. Plik konfiguracyjny Zabbix agent znajduje się w `C:\Zabbix\zabbix-8.0.0\conf`.

Aby uruchomić agent, skopiuj `zabbix_agentd.exe` i jego plik konfiguracyjny do dedykowanego folderu (np. `C:\Zabbix\agent`), a następnie uruchom 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
```

Instalacja agent na macOS z pakietu PKG

Przegląd

agent Zabbix można zainstalować w systemie macOS za pomocą pakietów instalacyjnych PKG, dostępnych do pobrania.

Pakiety agenta Zabbix są dostępne z szyfrowaniem lub bez niego.

Instalacja agent

Agent można zainstalować za pomocą graficznego interfejsu użytkownika lub z wiersza poleceń, na przykład:

```
sudo installer -pkg zabbix_agent-8.0.0-macos-arm64-openssl.pkg -target /
```

Upewnij się, że w poleceniu używasz poprawnej wersji pakietu Zabbix. Musi ona odpowiadać nazwie pobranego pakietu.

Uruchamianie agent

Agent uruchomi się automatycznie po instalacji lub restarcie.

W razie potrzeby możesz edytować plik konfiguracyjny `/usr/local/etc/zabbix/zabbix_agentd.conf`.

Aby uruchomić agent ręcznie, możesz wykonać:

```
sudo launchctl start com.zabbix.zabbix_agentd
```

Aby zatrzymać agent ręcznie:

```
sudo launchctl stop com.zabbix.zabbix_agentd
```

Podczas aktualizacji istniejący plik konfiguracyjny nie jest nadpisywany. Zamiast tego tworzony jest nowy plik `zabbix_agentd.conf.NEW`, który służy do przeglądu i aktualizacji istniejącego pliku konfiguracyjnego, jeśli to konieczne. Pamiętaj, aby po ręcznych zmianach w pliku konfiguracyjnym ponownie uruchomić agent.

Rozwiązywanie problemów i usuwanie agent

W tej sekcji wymieniono kilka przydatnych poleceń, których można użyć do rozwiązywania problemów i usuwania instalacji Zabbix agent.

Sprawdź, czy Zabbix agent jest uruchomiony:

```
ps aux | grep zabbix_agentd
```

Sprawdź, czy Zabbix agent został zainstalowany z pakietów:

```
pkgutil --pkgs | grep zabbix
com.zabbix.pkg.ZabbixAgent
```

Sprawdź pliki, które zostały zainstalowane z pakietu instalacyjnego (zwróć uwagę, że początkowy znak / nie jest wyświetlany w tym widoku):

```
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
```

Zatrzymaj Zabbix agent, jeśli został uruchomiony za pomocą `launchctl`:

```
sudo launchctl unload /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
```

Usuń pliki (w tym konfigurację i logi), które zostały zainstalowane przez pakiet instalacyjny:

```
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
```

Usuń informację o tym, że Zabbix agent został zainstalowany:

```
sudo pkgutil --forget com.zabbix.pkg.ZabbixAgent
```

Building Zabbix agent on macOS

Przegląd

Ta sekcja pokazuje, jak zbudować binaria agenta Zabbix dla systemu macOS ze źródeł, z obsługą TLS lub bez niej.

Wymagania wstępne

Będziesz potrzebować narzędzi deweloperskich wiersza poleceń (Xcode nie jest wymagany), Automake, pkg-config oraz PCRE (v8.x) lub PCRE2 (v10.x). Jeśli chcesz zbudować pliki binarne agent z obsługą TLS, będziesz także potrzebować OpenSSL lub GnuTLS.

Aby zainstalować Automake i pkg-config, będziesz potrzebować menedżera pakietów Homebrew ze strony https://brew.sh/. Aby go zainstalować, otwórz terminal i uruchom następujące polecenie:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Następnie zainstaluj Automake i pkg-config:

```
brew install automake
brew install pkg-config
```

Przygotowanie bibliotek PCRE, OpenSSL i GnuTLS zależy od sposobu, w jaki będą one linkowane z agent.

Jeśli zamierzasz uruchamiać pliki binarne agent na komputerze z macOS, który ma już te biblioteki, możesz użyć wstępnie skompilowanych bibliotek dostarczanych przez Homebrew. Zwykle są to komputery z macOS, które używają Homebrew do budowania plików binarnych Zabbix agent lub do innych celów.

Jeśli pliki binarne agent będą używane na komputerach z macOS, które nie mają współdzielonej wersji bibliotek, powinieneś skompilować biblioteki statyczne ze źródeł i połączyć z nimi Zabbix agent.

Budowanie plików binarnych agent z bibliotekami współdzielonymi

Zainstaluj PCRE2 (w razie potrzeby zastąp *pcre2* przez *pcre* w poniższych poleceniach):

```
brew install pcre2
```

Podczas budowania z TLS zainstaluj OpenSSL i/lub GnuTLS:

```
brew install openssl
brew install gnutls
```

Pobierz kod źródłowy Zabbix:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

Zbuduj agent bez TLS:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6
make
make install
```

Zbuduj agent z OpenSSL:

```
cd zabbix
./bootstrap.sh
./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-openssl=/usr/local/opt/
make
make install
```

Zbuduj agent z 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
```

Budowanie plików binarnych agent z bibliotekami statycznymi bez TLS

Załóżmy, że statyczne biblioteki PCRE zostaną zainstalowane w `$HOME/static-libs`. Użyjemy PCRE2 10.39.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
```

Pobierz i zbuduj PCRE z obsługą właściwości 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
```

Pobierz kod źródłowy Zabbix i zbuduj agent:

```
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
```

Budowanie plików binarnych agent z bibliotekami statycznymi przy użyciu OpenSSL

Podczas budowania OpenSSL zaleca się uruchomienie `make test` po pomyślnym zakończeniu kompilacji. Nawet jeśli kompilacja zakończyła się sukcesem, testy czasami kończą się niepowodzeniem. W takim przypadku należy zbadać problemy i je rozwiązać przed kontynuowaniem.

Załóżmy, że statyczne biblioteki PCRE i OpenSSL zostaną zainstalowane w `$HOME/static-libs`. Użyjemy PCRE2 10.39 oraz OpenSSL 1.1.1a.

```
PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
OPENSSL_PREFIX="$HOME/static-libs/openssl-1.1.1a"
```

Zbudujmy biblioteki statyczne w `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Pobierz i zbuduj PCRE z obsługą właściwości 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 ..
```

Pobierz i zbuduj 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 ..
```

Pobierz kod źródłowy Zabbix i zbuduj agent:

```
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
```

Budowanie plików binarnych agent z bibliotekami statycznymi przy użyciu GnuTLS

GnuTLS zależy od backendu kryptograficznego Nettle oraz biblioteki arytmetycznej GMP. Zamiast używać pełnej biblioteki GMP, ten przewodnik wykorzystuje mini-gmp, które jest dołączone do Nettle.

Podczas budowania GnuTLS i Nettle zaleca się uruchomienie `make check` po pomyślnym zakończeniu kompilacji. Nawet jeśli kompilacja zakończyła się powodzeniem, testy czasami kończą się niepowodzeniem. W takim przypadku należy zbadać problemy

i rozwiązać je przed kontynuowaniem.

Załóżmy, że statyczne biblioteki PCRE, Nettle i GnuTLS zostaną zainstalowane w `$HOME/static-libs`. Użyjemy PCRE2 10.39, Nettle 3.4.1 oraz 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"
```

Zbudujmy biblioteki statyczne w `static-libs-source`:

```
mkdir static-libs-source
cd static-libs-source
```

Pobierz i zbuduj 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 ..
```

Pobierz i zbuduj 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 ..
```

Pobierz kod źródłowy Zabbix i zbuduj agent:

```
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
```

Przewodnik wdrażania serwera Zabbix w AWS

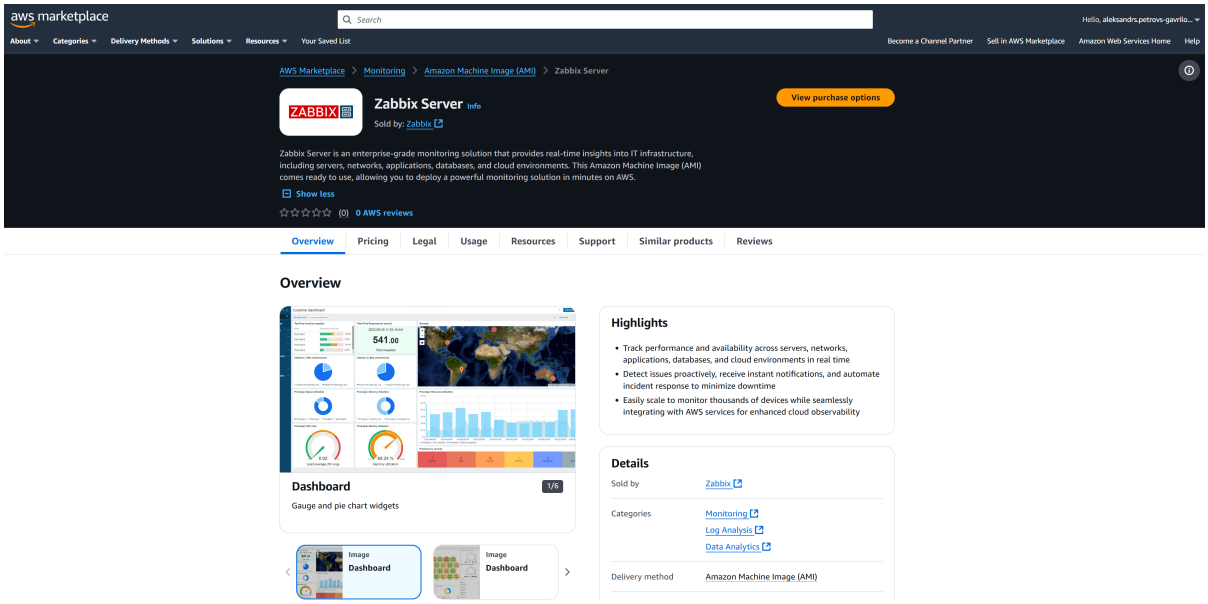
Przegląd

Ta strona opisuje sposób wdrażania serwera Zabbix przy użyciu AWS.

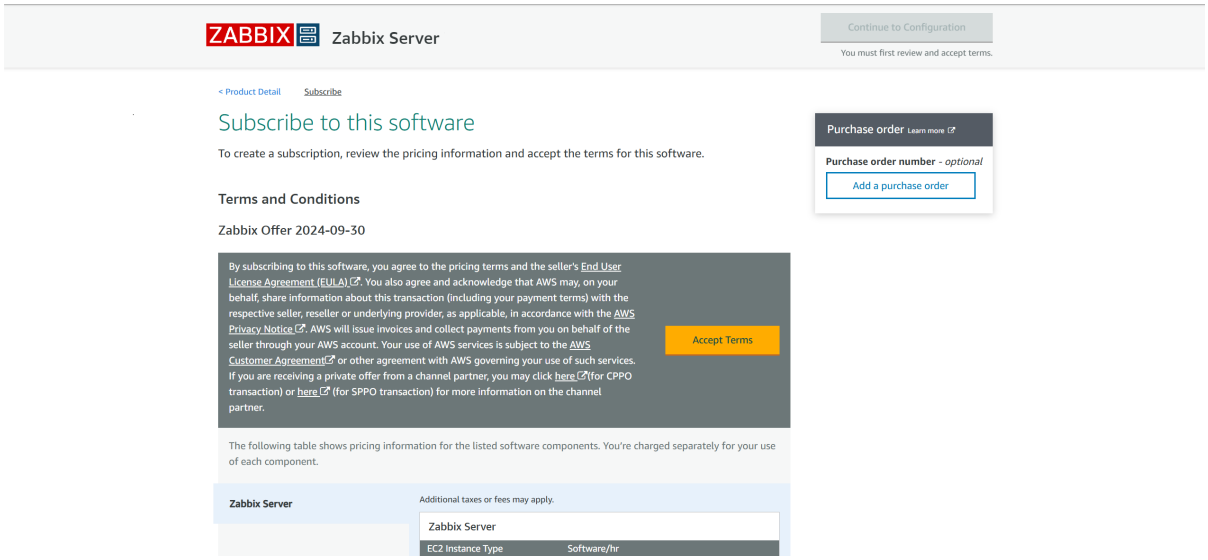
> **Attention:**
> Na platformie dostępnych jest kilka wersji Zabbix do wdrożenia. Upewnij się, że czytasz dokumentację dotyczącą tej samej wersji Zabbix, którą wdrożono. Dokumentację dotyczącą różnych wersji Zabbix można wybrać z menu rozwijanego wersji.

Wdrażanie appliance

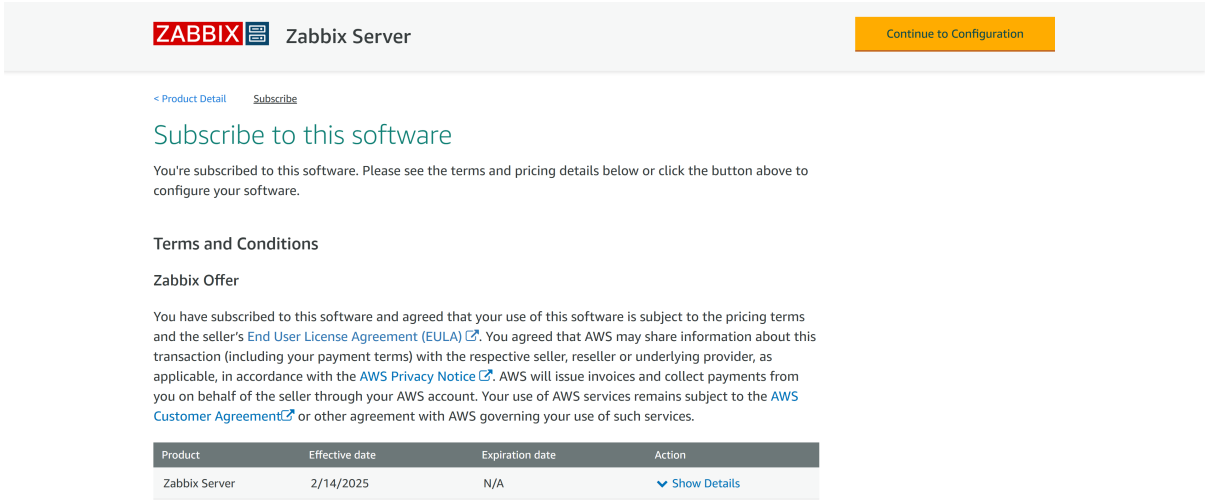
Aby wdrożyć appliance Zabbix, najpierw przejdź do strony Zabbix w AWS Marketplace. Następnie kliknij opcję *View purchase options*, aby wdrożyć obraz i przejść do strony subskrypcji.



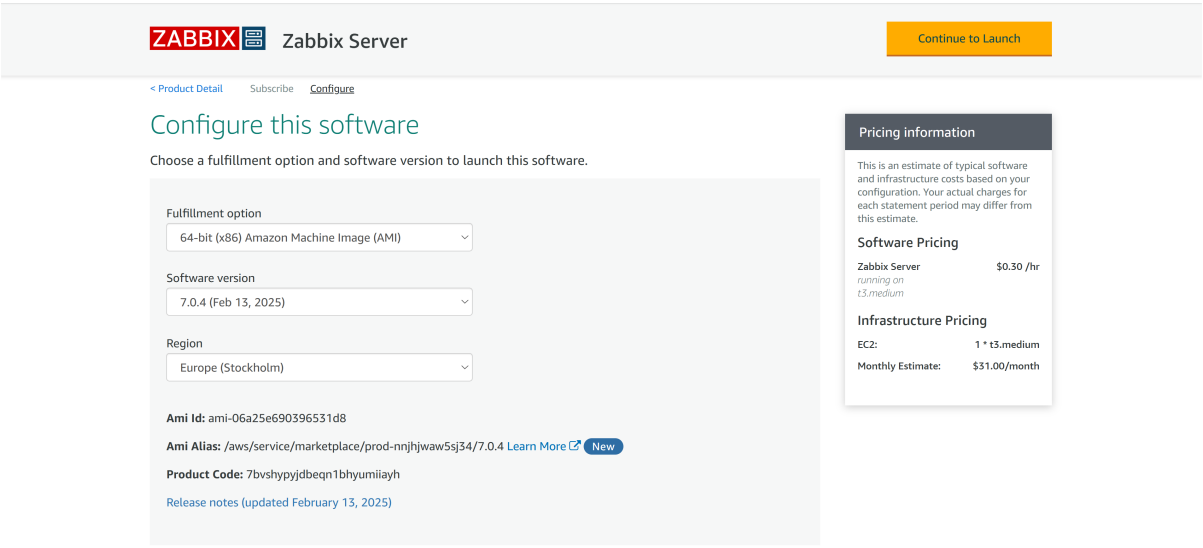
Kliknij przycisk *Accept terms*, aby zaakceptować *Terms and Conditions*.



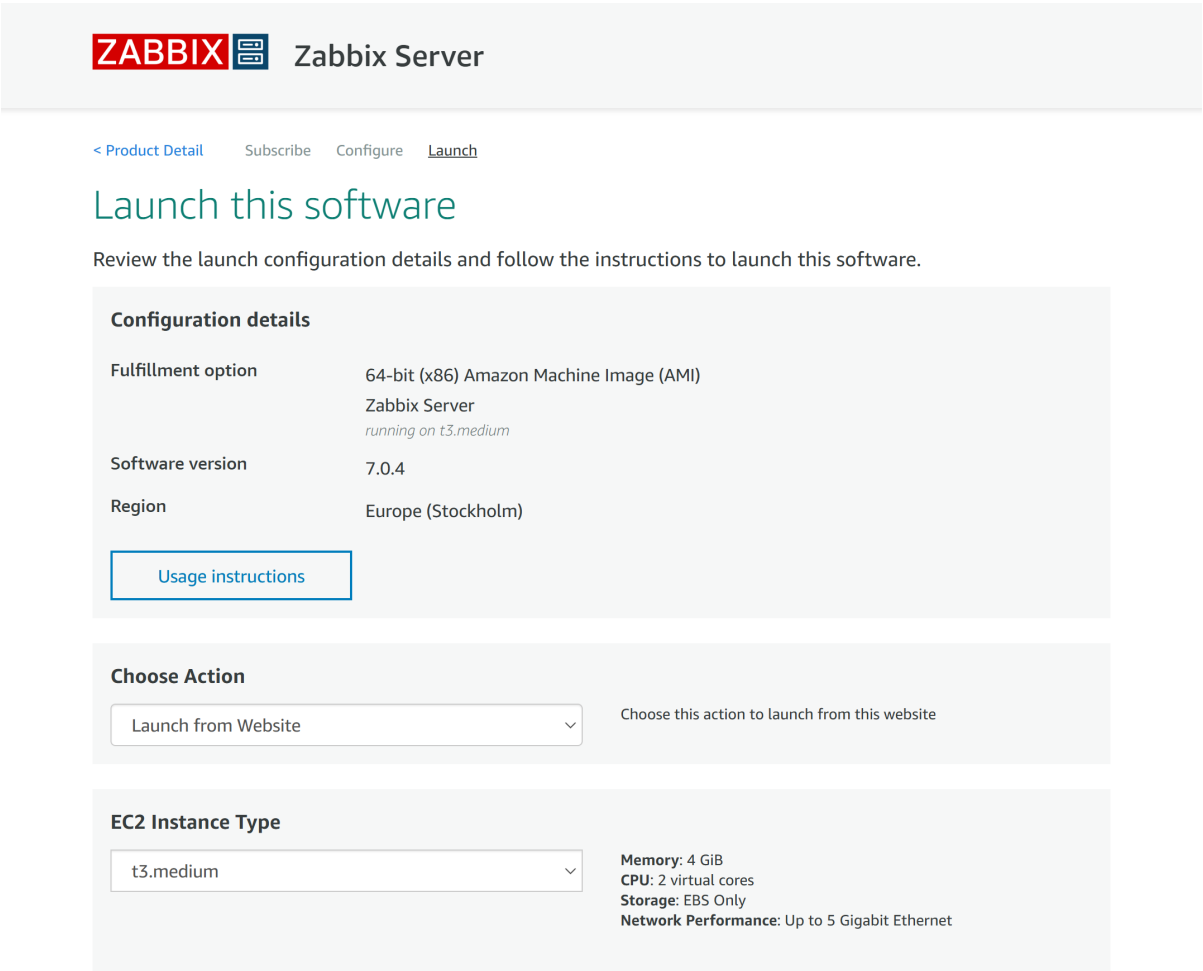
Kliknij przycisk *Continue to Configuration*, aby przejść do sekcji Configuration dla appliance Zabbix.



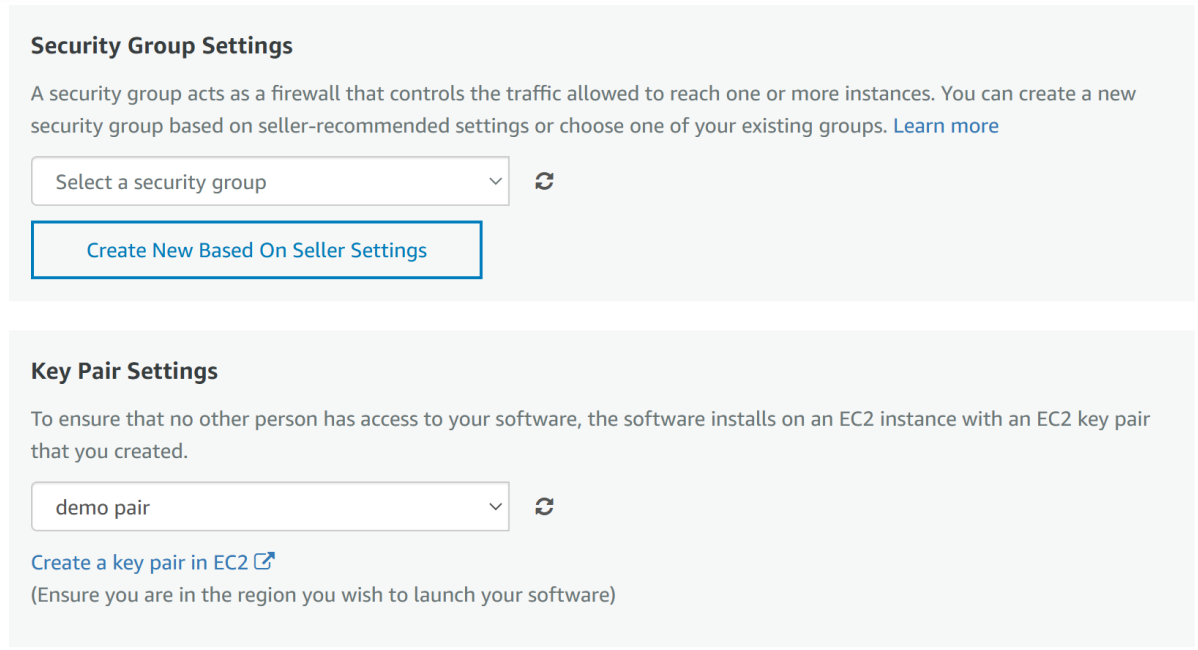
Wybierz opcję Fulfilment, wersję Zabbix oraz preferowany Region.



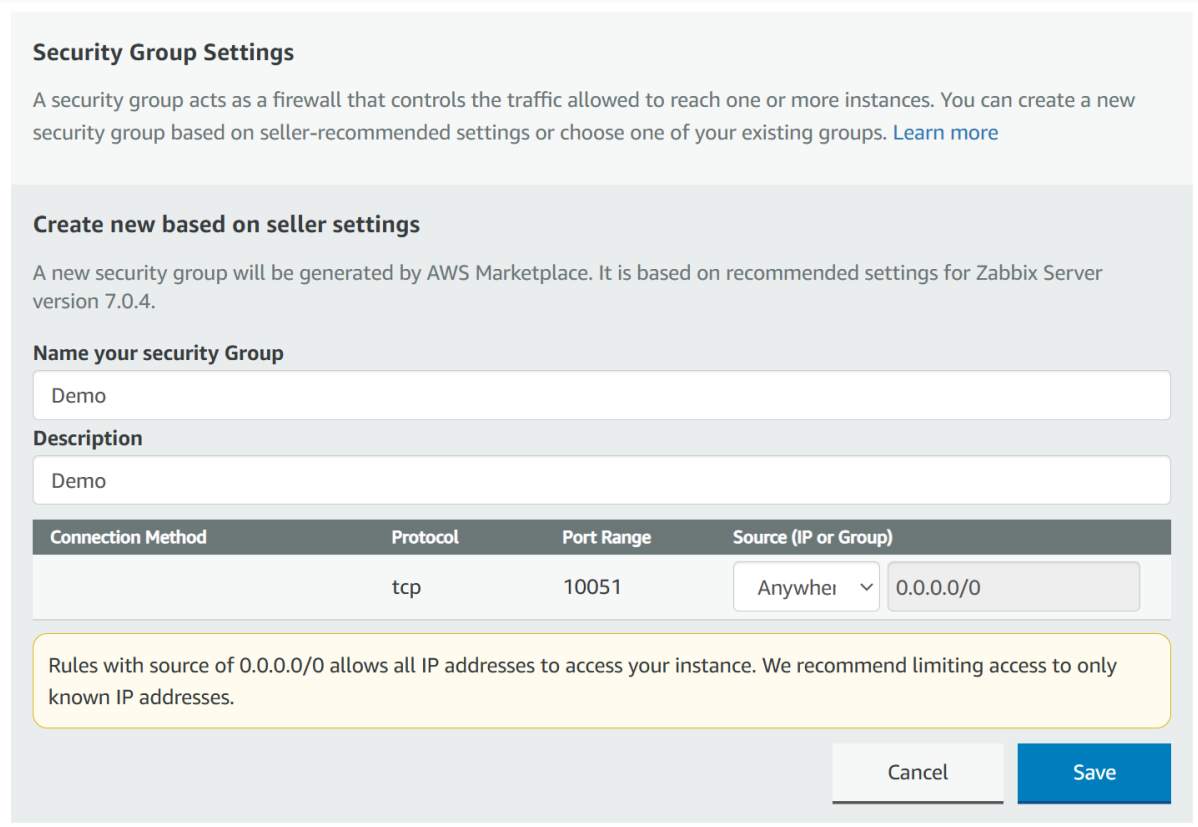
Kliknij przycisk *Continue to Launch*. Na stronie Launch wybierz preferowaną opcję Launch, typ instancji EC2 oraz ustawienia VPC i Subnet.



W dolnej części strony Launch określ ustawienia Security Group i wybierz lub utwórz parę kluczy EC2 — pamiętaj, aby zapisać klucz prywatny w bezpiecznej lokalizacji!



Jeśli wybierzesz opcję *Create a new Security Group based on seller settings*, zostanie zezwolony ruch na porcie TCP 10051 (serwer Zabbix). Uwaga: jeśli chcesz uzyskać dostęp do instancji za pośrednictwem SSH oraz HTTP/HTTPS (frontend), musisz również zezwolić na połączenia SSH i HTTP/HTTPS w ustawieniach Security Group.



Gratulacje, appliance Zabbix został pomyślnie wdrożony.

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)

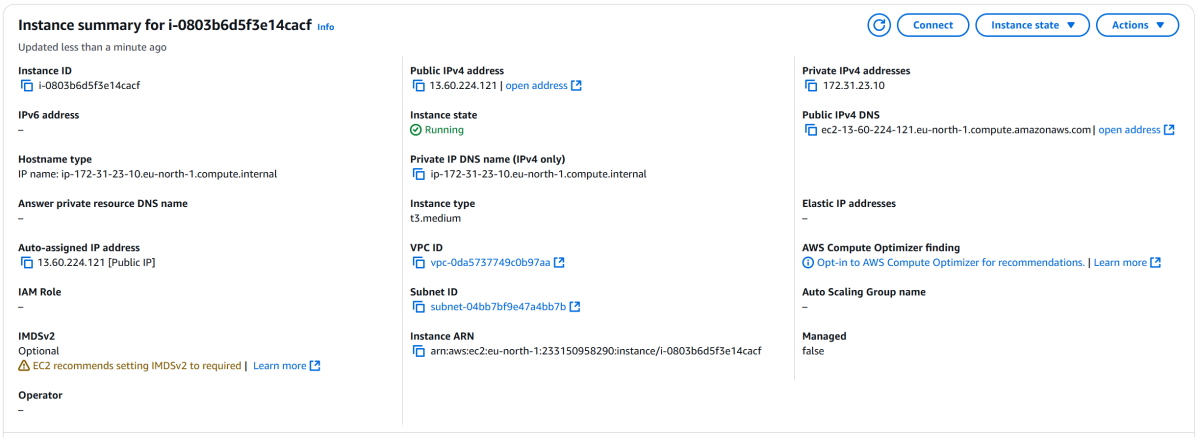
Usage instructions

Choose Action

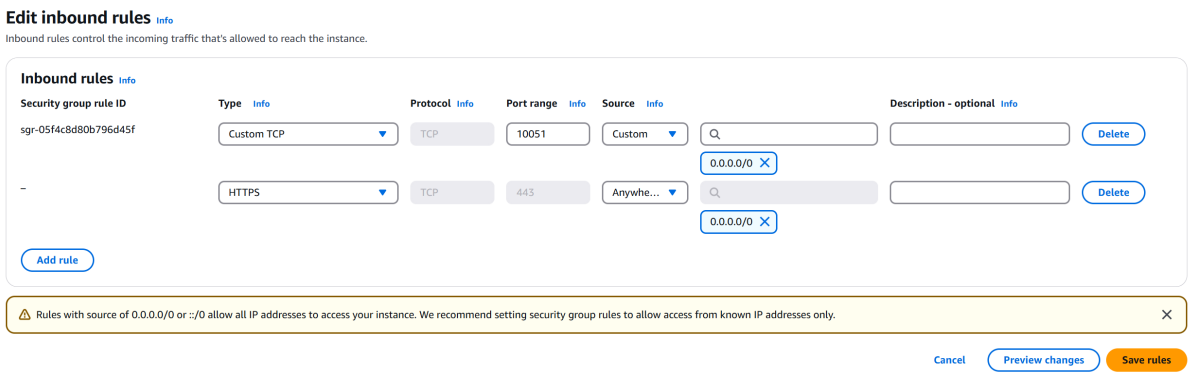
Select a launch action

Uzyskiwanie dostępu do appliance

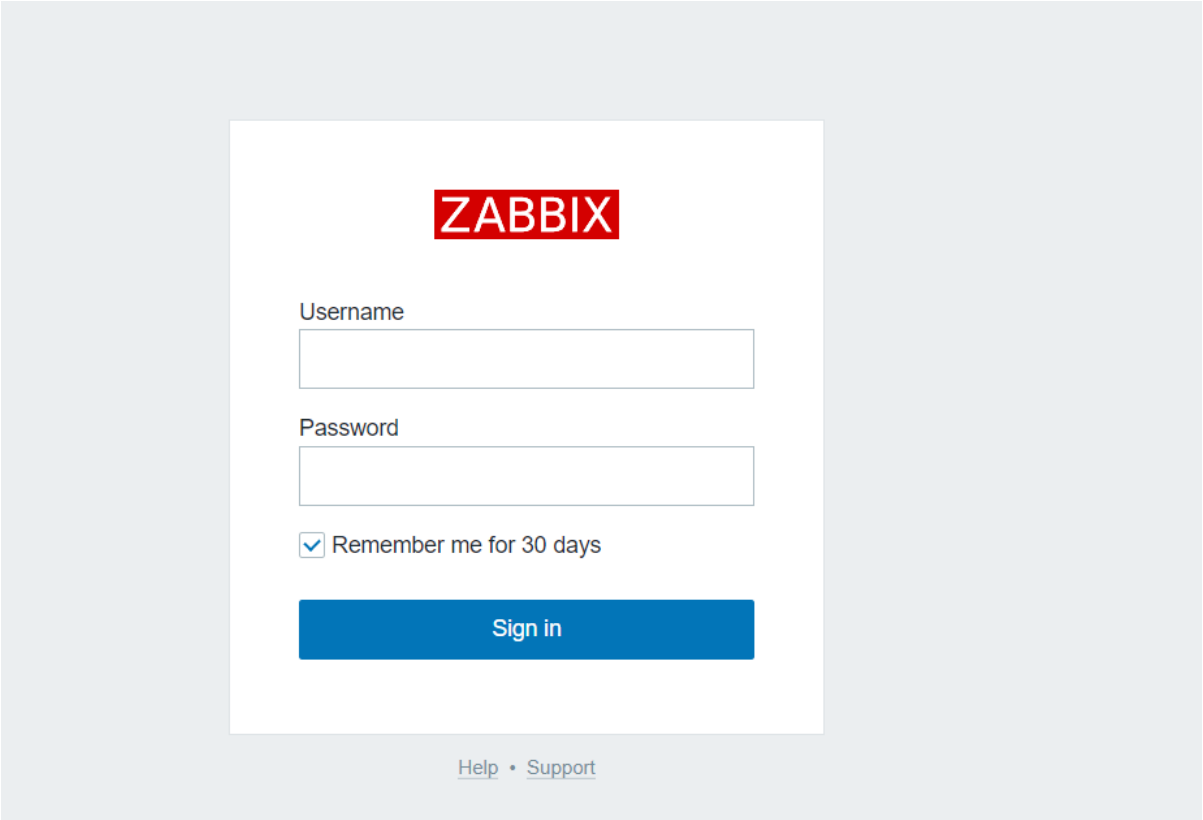
Otwórz sekcję *Instances* i wybierz nowo wdrożoną instancję Zabbix appliance.



Aby uzyskać dostęp do frontend Zabbix za pomocą publicznego adresu IPv4 lub publicznego DNS IPv4 instancji, zezwól na przychodzący ruch HTTPS na porcie 443 w Security Group.



Teraz możemy uzyskać dostęp do frontend Zabbix, otwierając publiczny adres IPv4 lub publiczny DNS IPv4 Zabbix appliance.



Aby uzyskać hasło, musimy połączyć się z maszyną za pomocą konsoli szeregowej lub SSH. Domyślna nazwa użytkownika SSH to rocky. Nie zapomnij użyć pary kluczy wybranej lub wygenerowanej wcześniej. Pamiętaj, że generowanie danych uwierzytelniających może zająć trochę czasu. Generowanie danych uwierzytelniających może potrwać kilka minut. Odczekaj kilka minut przed próbą nawiązania połączenia.

```
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

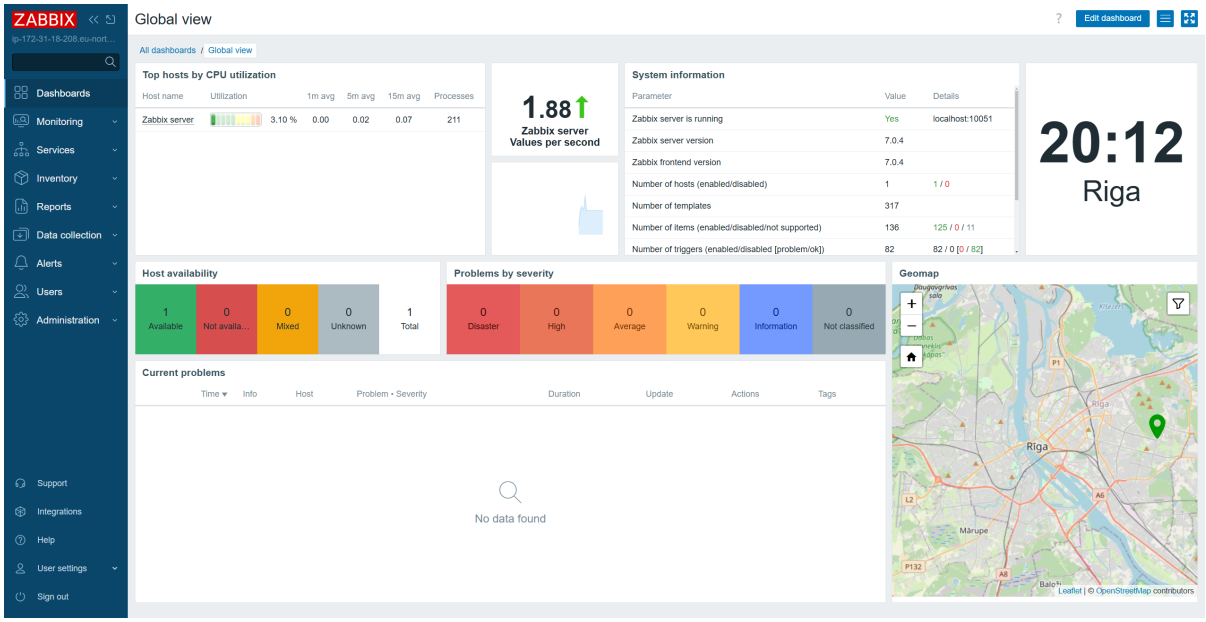
To learn about available professional services, including technical suppport and training, please visit https://www.zabbix.com/services

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.

********************************************************************************
```

Teraz możesz użyć tych informacji, aby zalogować się do frontend Zabbix.



Uzyskiwanie dostępu do bazy danych

Do bazy danych Zabbix można uzyskać dostęp za pomocą użytkownika `root` bez podawania hasła.

```
[root@ip-172-31-29-138 ~]# mysql -uroot
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 124
Server version: 8.0.26 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Dane uwierzytelniające są przechowywane w pliku konfiguracyjnym `/root/.my.cnf`.

Domyślną strefę czasową frontend Zabbix można zmienić w sekcji *Administracja* > *Ogólne* > *GUI*

Przewodnik wdrażania proxy Zabbix w AWS

Przegląd

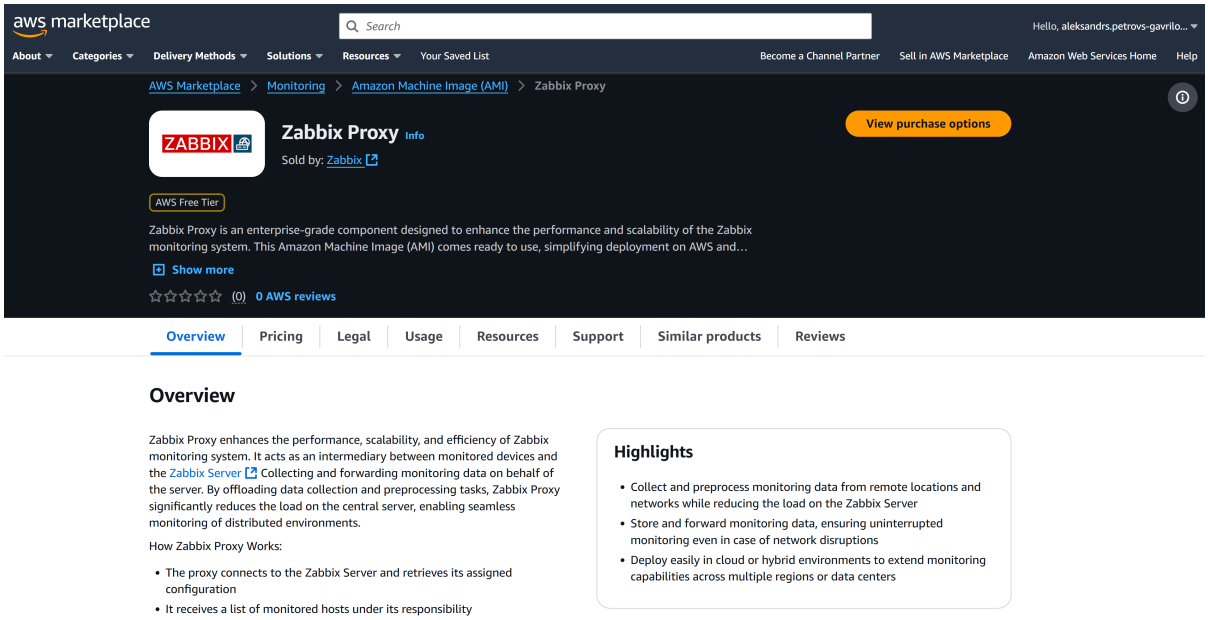
Ta strona opisuje sposób wdrażania proxy Zabbix przy użyciu AWS.

Attention:
Na platformie dostępnych jest kilka wersji Zabbix do wdrożenia. Upewnij się, że czytasz dokumentację dotyczącą tej samej wersji Zabbix, którą wdrożono. Dokumentację dotyczącą różnych wersji Zabbix można wybrać z menu rozwijanego wersji.

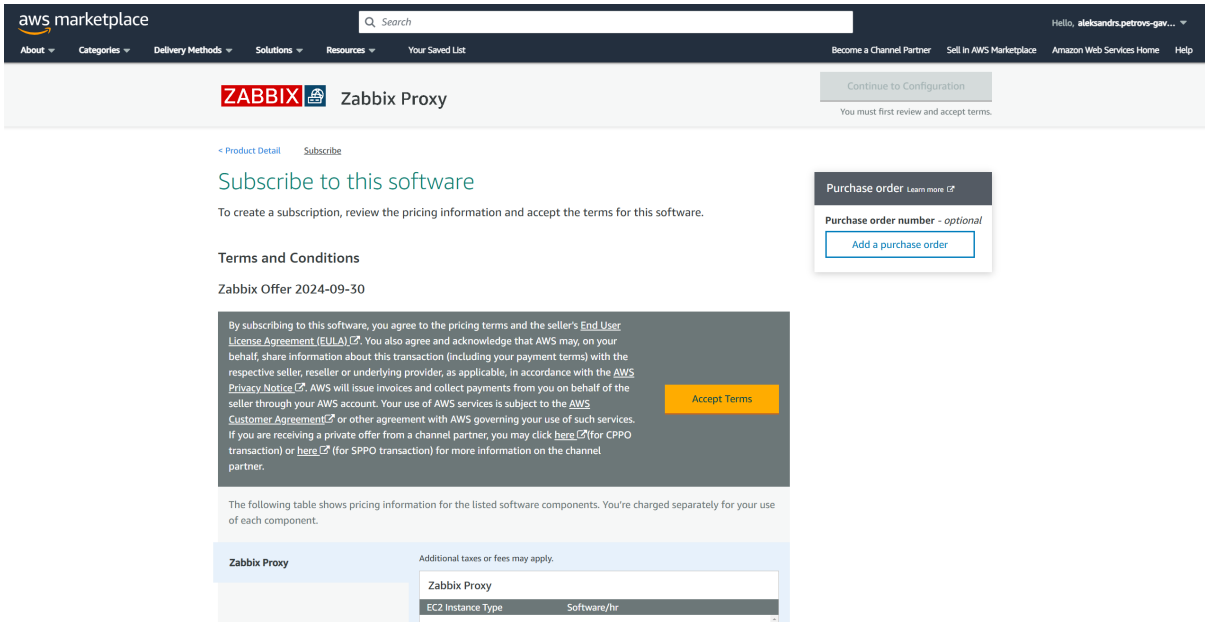
Wdrażanie proxy Zabbix

Attention:
Do korzystania z komponentu proxy Zabbix wymagana jest działająca instancja serwera Zabbix.

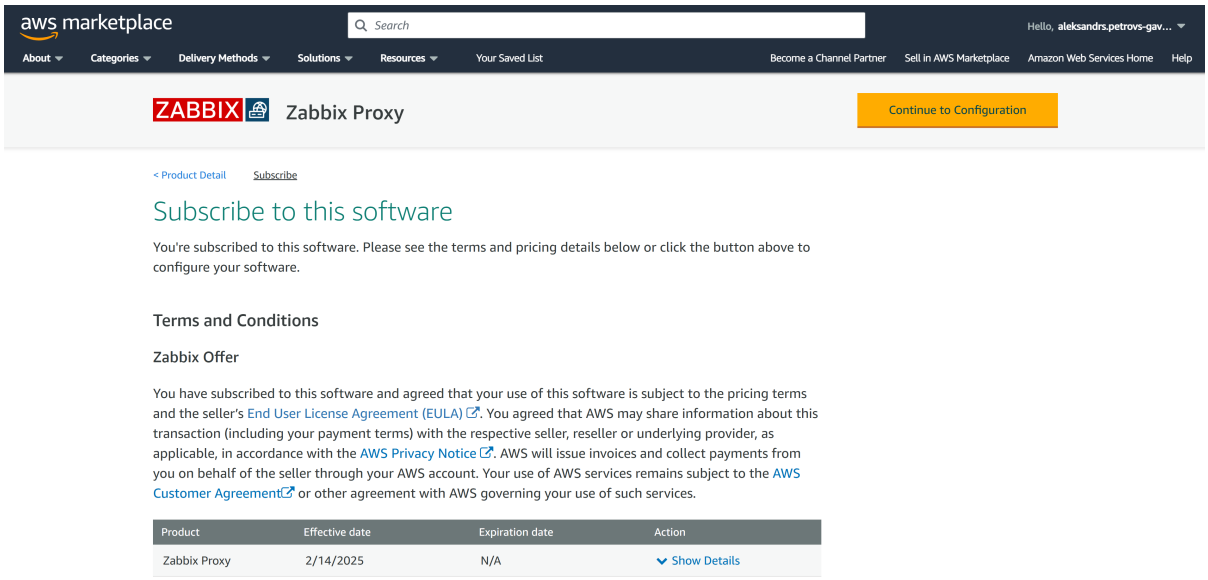
Aby wdrożyć proxy Zabbix, najpierw przejdź do strony Zabbix w AWS Marketplace. Następnie kliknij *View purchase options*, aby wdrożyć obraz i przejść do strony subskrypcji.



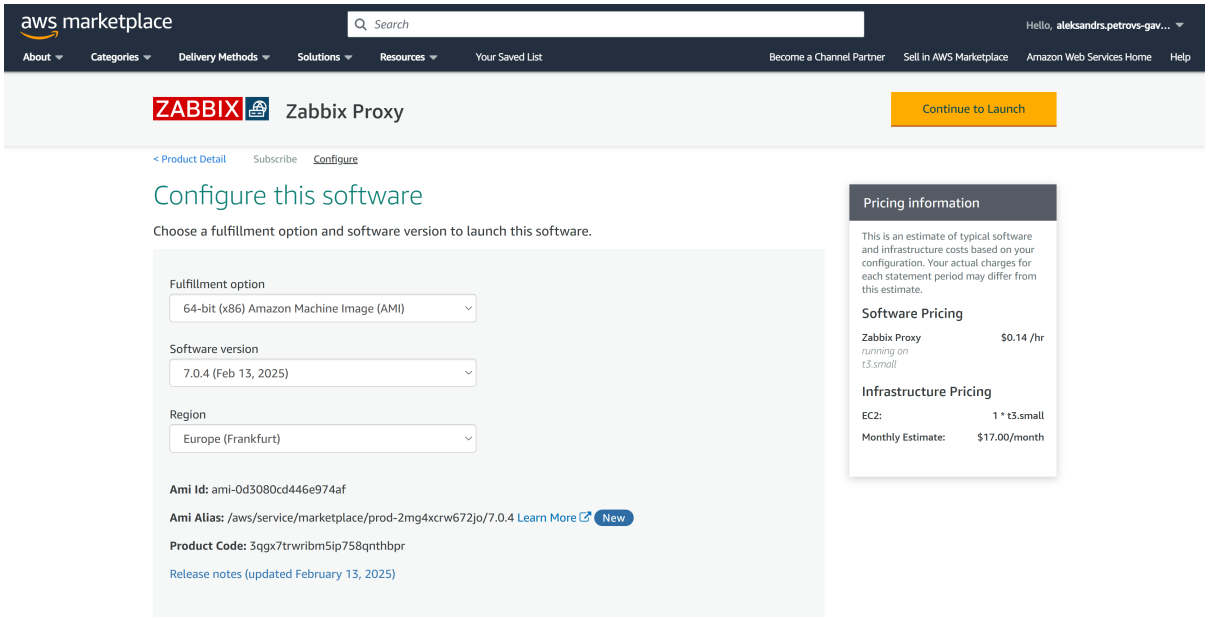
Kliknij przycisk *Accept terms*, aby zaakceptować *Terms and Conditions*.



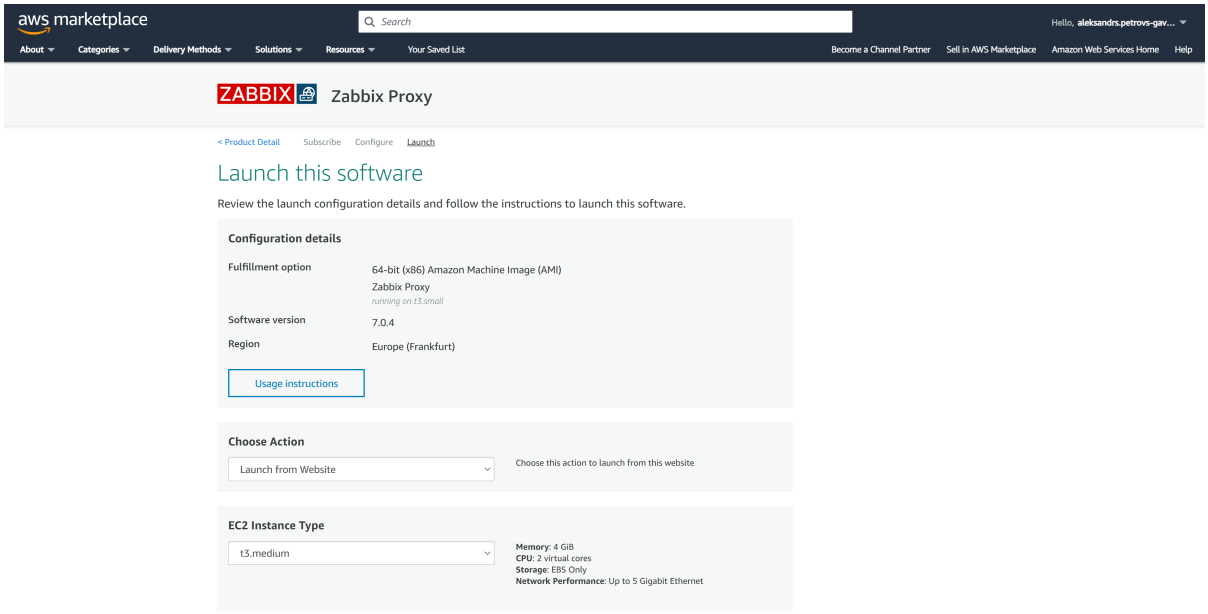
Kliknij przycisk *Continue to Configuration*, aby przejść do sekcji Configuration dla proxy Zabbix.



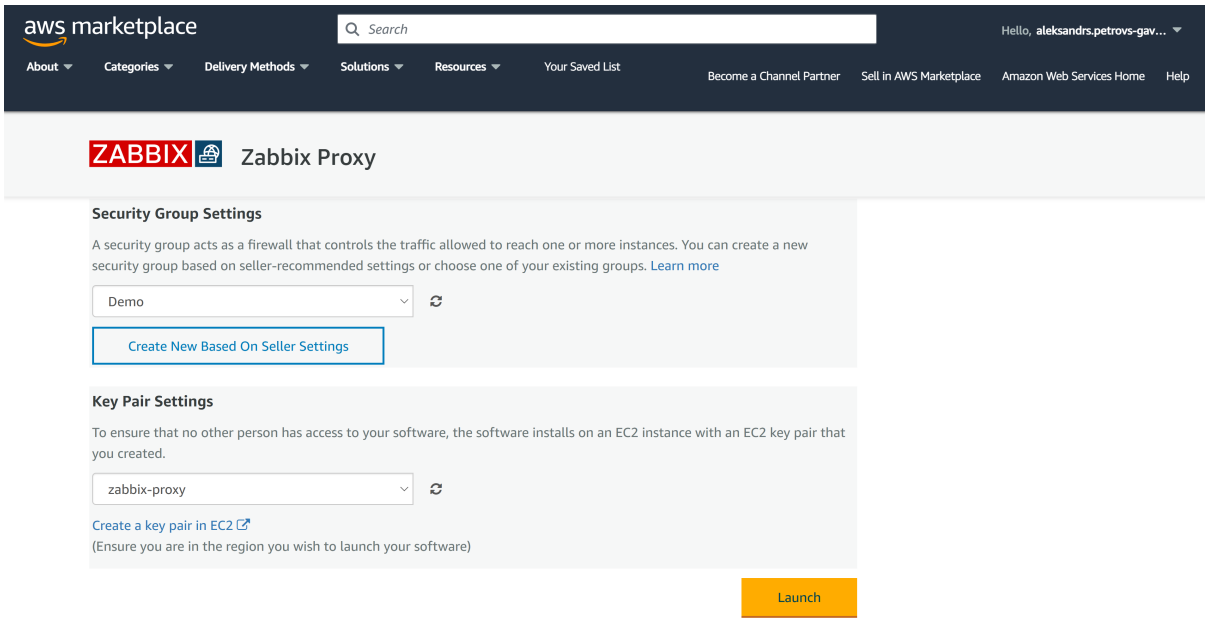
Wybierz opcję Fulfilment, wersję Zabbix oraz preferowany Region.



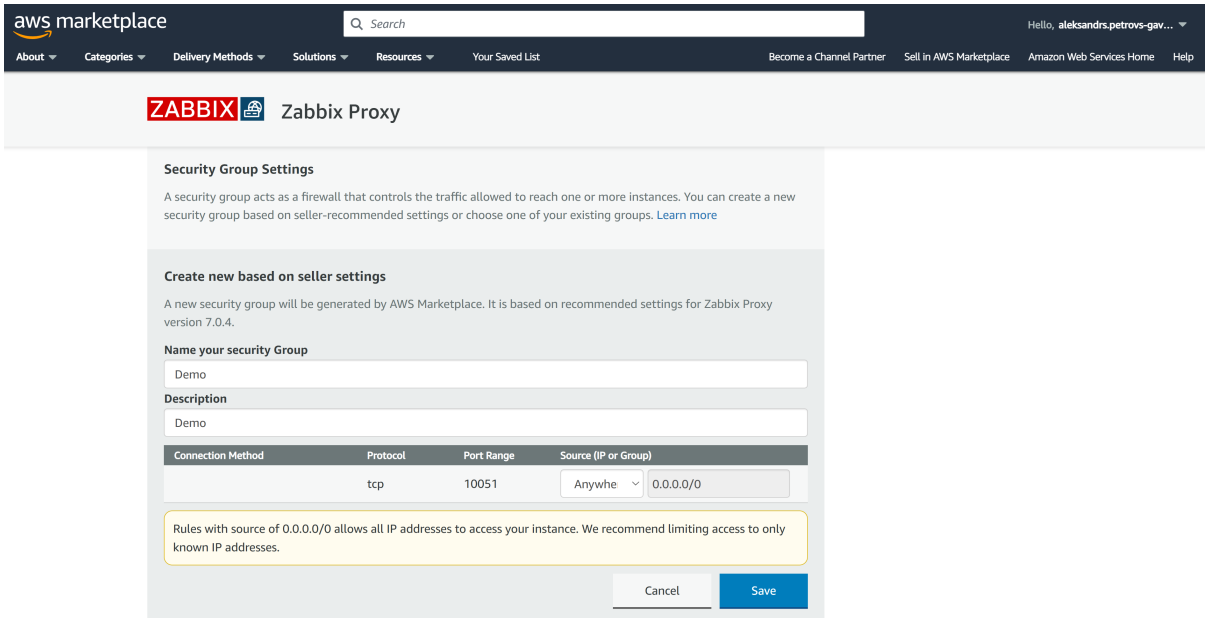
Kliknij przycisk *Continue to Launch*. Na stronie Launch wybierz preferowaną opcję Launch, typ instancji EC2 oraz ustawienia VPC i Subnet.



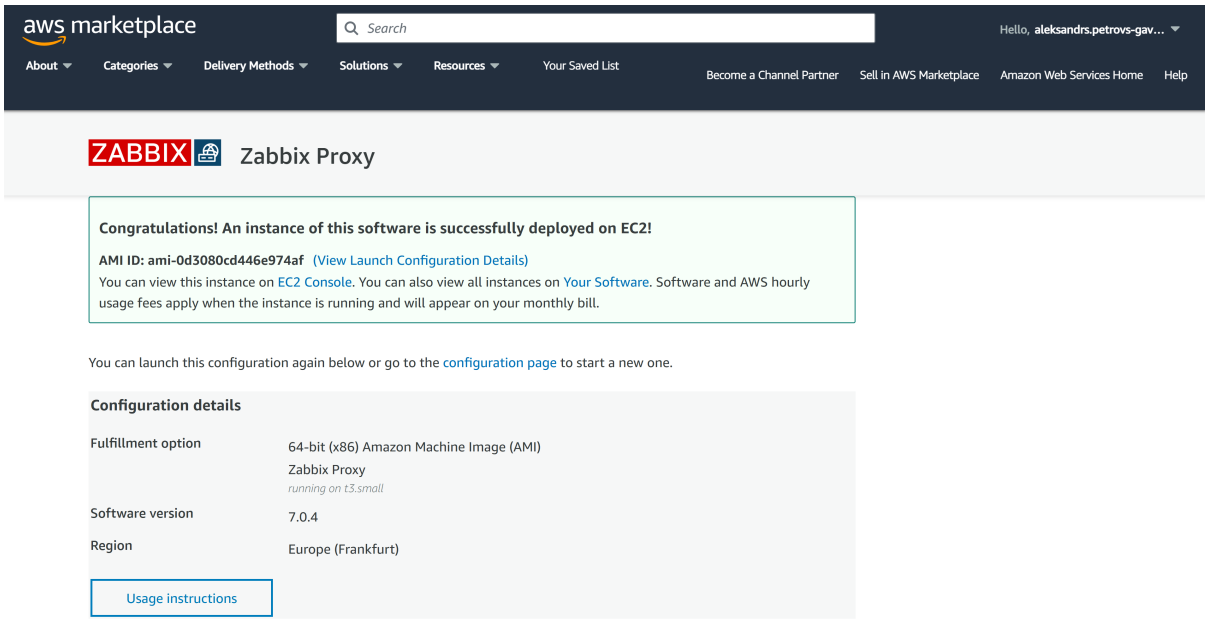
Na dole strony Launch określ ustawienia Security Group oraz wybierz lub utwórz parę kluczy EC2. Pamiętaj, aby zapisać klucz prywatny w bezpiecznej lokalizacji.



Jeśli wybierzesz opcję *Create a new Security Group based on the Seller settings*, zostanie dozwolony port TCP 10051 (serwer Zabbix). Uwaga: jeśli chcesz uzyskiwać dostęp do instancji za pośrednictwem SSH, musisz również zezwolić na połączenia SSH w ustawieniach Security Group.

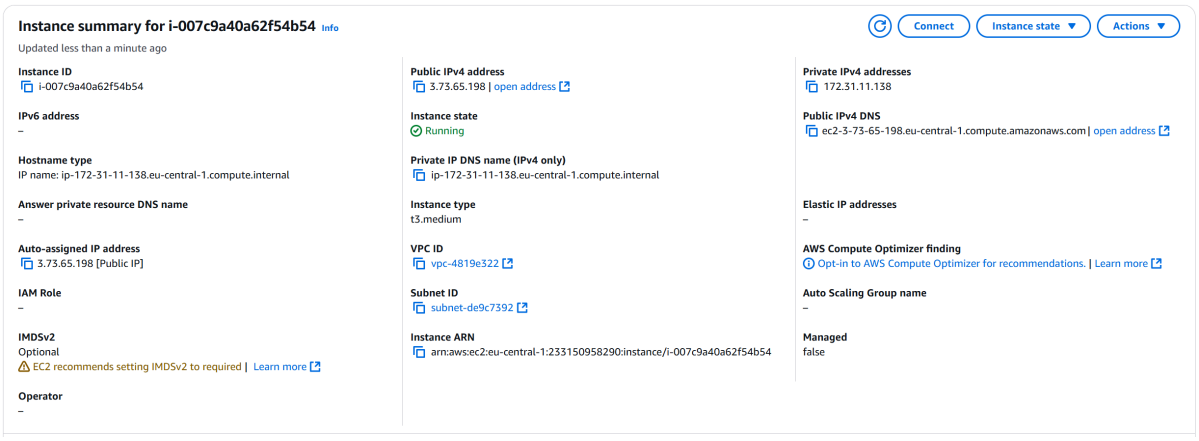


Na koniec kliknij przycisk *Launch*. Gratulacje, proxy Zabbix jest teraz wdrażane.



Konfigurowanie i uzyskiwanie dostępu do proxy Zabbix

Otwórz sekcję Instances w konsoli EC2 i wybierz nowo wdrożoną instancję proxy Zabbix.



Połącz się z adresem Public IPv4 lub Public IPv4 DNS proxy Zabbix za pomocą SSH. Domślna nazwa użytkownika SSH to `rocky`. Nie zapomnij użyć pary kluczy wygenerowanej wcześniej.

Generowanie danych uwierzytelniających może potrwać kilka minut. Przed próbą nawiązania połączenia odczekaj kilka minut.



Plik konfiguracyjny proxy Zabbix znajduje się w `/etc/zabbix/zabbix_proxy.conf`.

Plik konfiguracyjny można otworzyć za pomocą wybranego edytora tekstu, na przykład vi:

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

Początkowa konfiguracja proxy polega na zmodyfikowaniu parametrów `ProxyMode`, `Server` i `Hostname` w pliku konfiguracyjnym proxy Zabbix.

Tryb proxy można zmienić, edytując parametr `ProxyMode`. Ustaw `ProxyMode=0` dla trybu aktywnego lub `ProxyMode=1` dla trybu pasywnego.

Parametr `Server` powinien zawierać adres IP serwera Zabbix. Na przykład:

```
Server=192.168.50.141
```

Parametr `Hostname` powinien zawierać nazwę hosta proxy Zabbix. Zdefiniowana tutaj nazwa hosta powinna odpowiadać nazwie proxy podanej podczas konfigurowania proxy w frontendzie Zabbix. Na przykład:

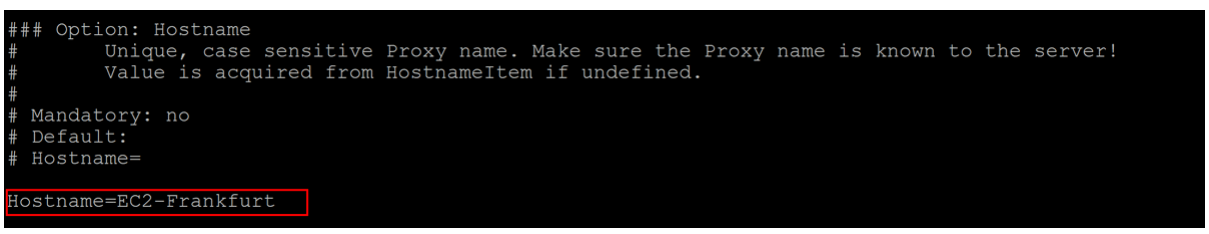
```
Hostname=EC2-Frankfurt
```

Po zakończeniu konfiguracji uruchom ponownie proxy Zabbix za pomocą następującego polecenia:

```
sudo systemctl restart zabbix-proxy
```

Uwaga: użyj sudo, aby edytować pliki konfiguracyjne i wykonywać polecenia z wiersza poleceń.



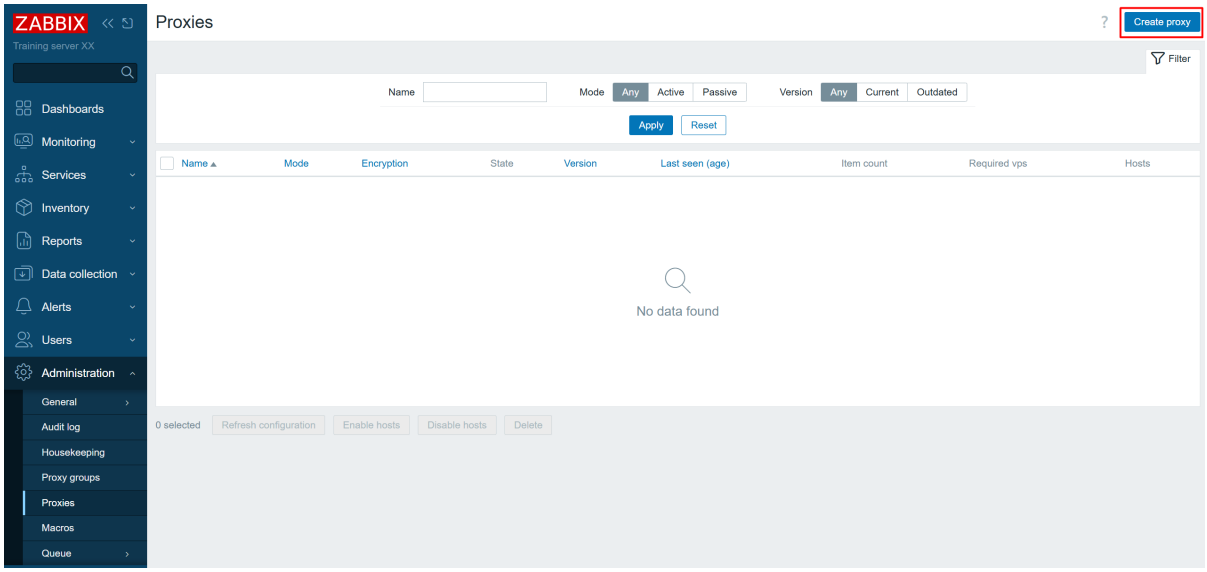


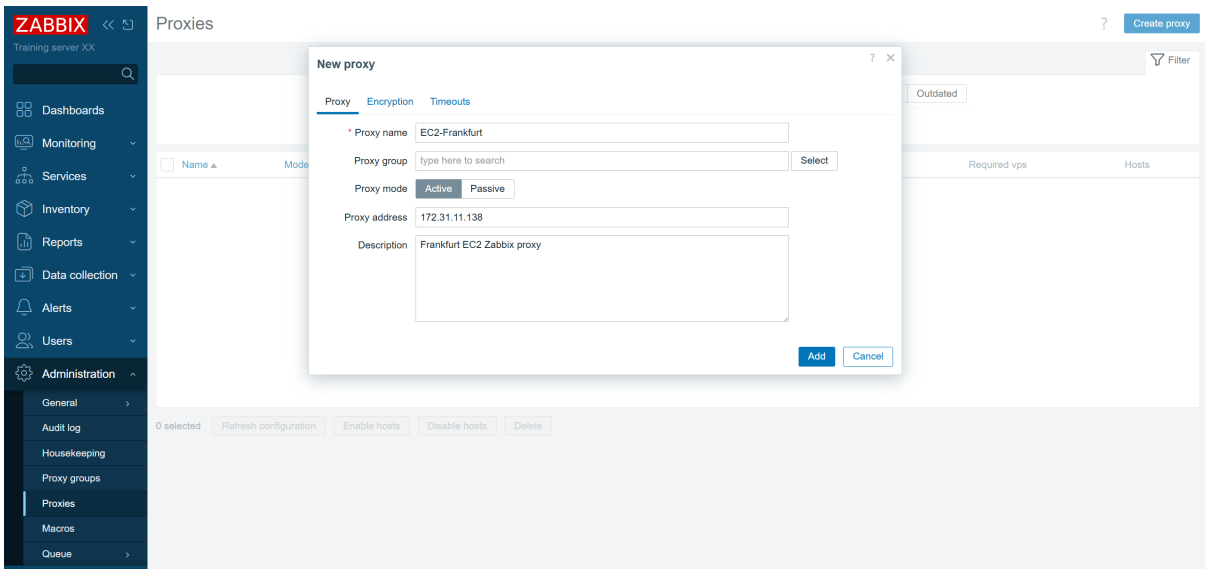
Otwórz frontend Zabbix i przejdź do *Administration* > *Proxies*. Kliknij przycisk *Create proxy*.

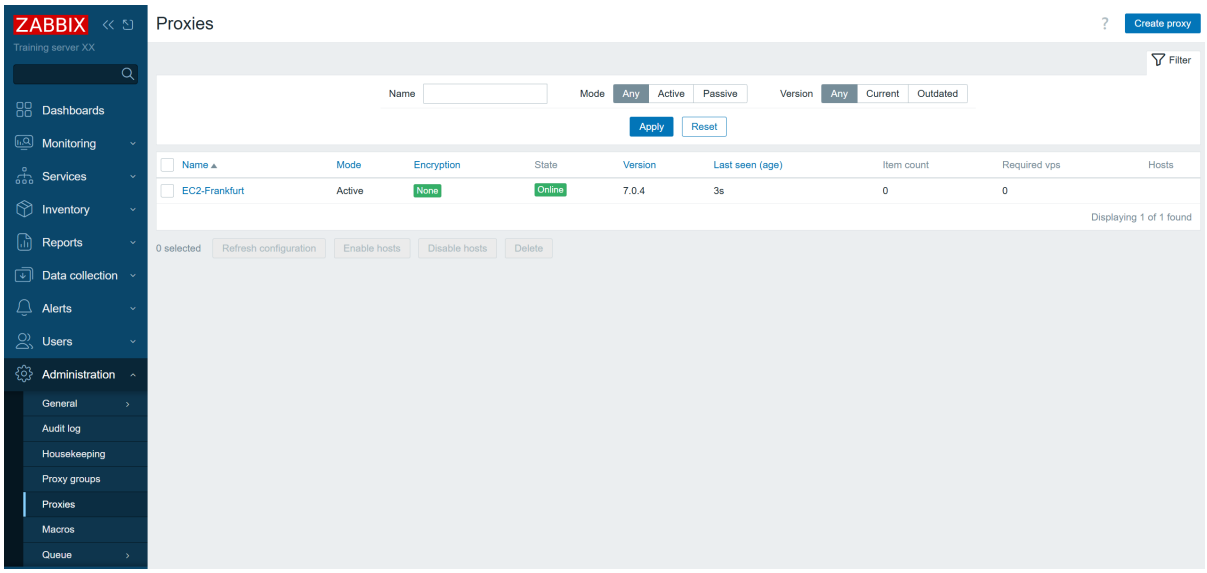
Wprowadź tę samą nazwę proxy co wartość parametru `Hostname` w pliku konfiguracyjnym proxy.

Wybierz tryb proxy i podaj adres IP proxy (opcjonalnie w przypadku proxy aktywnych).

Kliknij *Create proxy*.







Lokalizacja bazy danych proxy

Baza danych SQLite proxy jest przechowywana w `/tmp/zabbix_proxy.sqlite3` i określana przez parametr `DBName` w pliku konfiguracyjnym proxy Zabbix.

Przewodnik wdrażania serwera Zabbix na platformie Azure

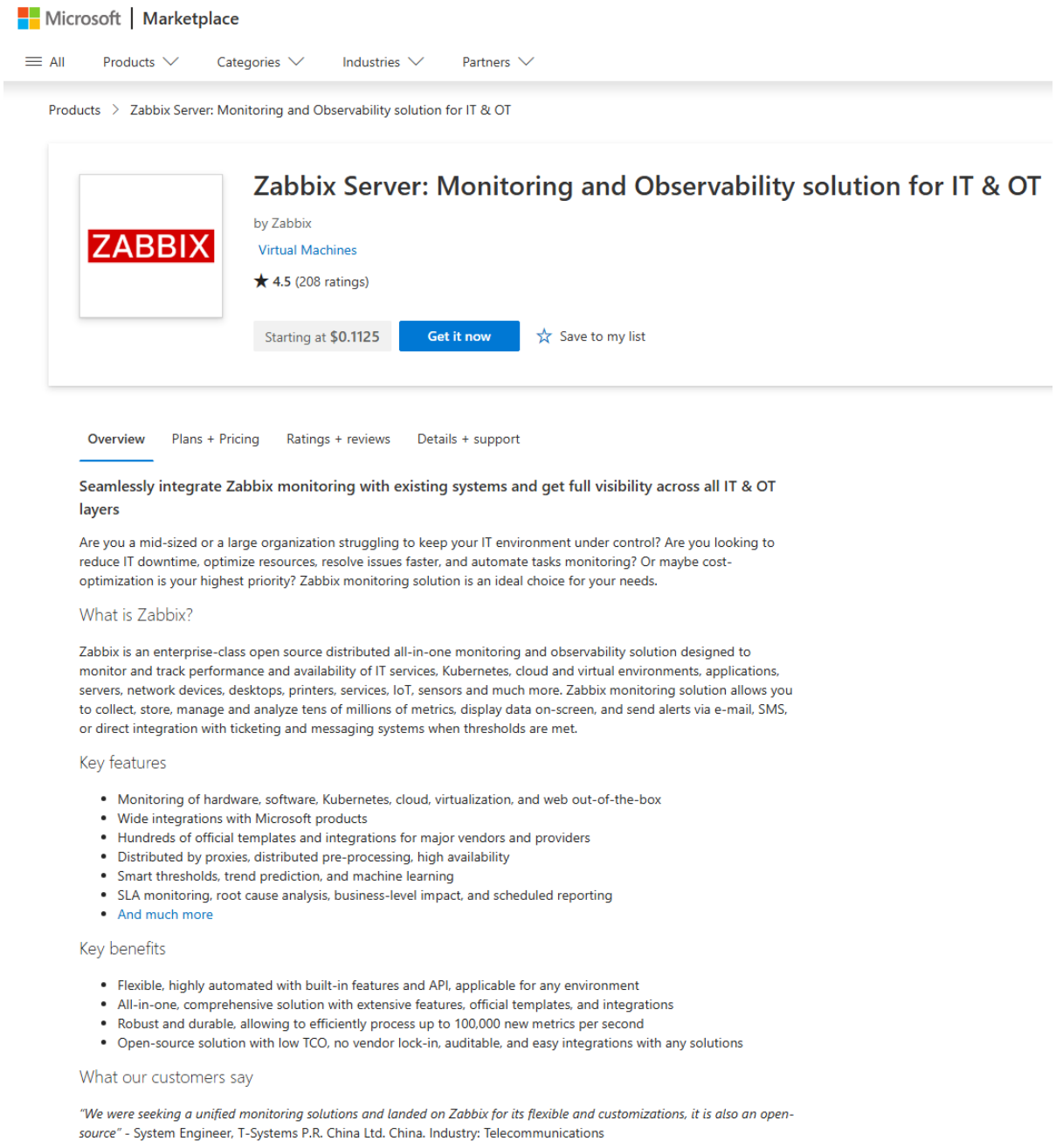
Przegląd

Ta strona opisuje sposób wdrażania serwera Zabbix przy użyciu platformy Azure.

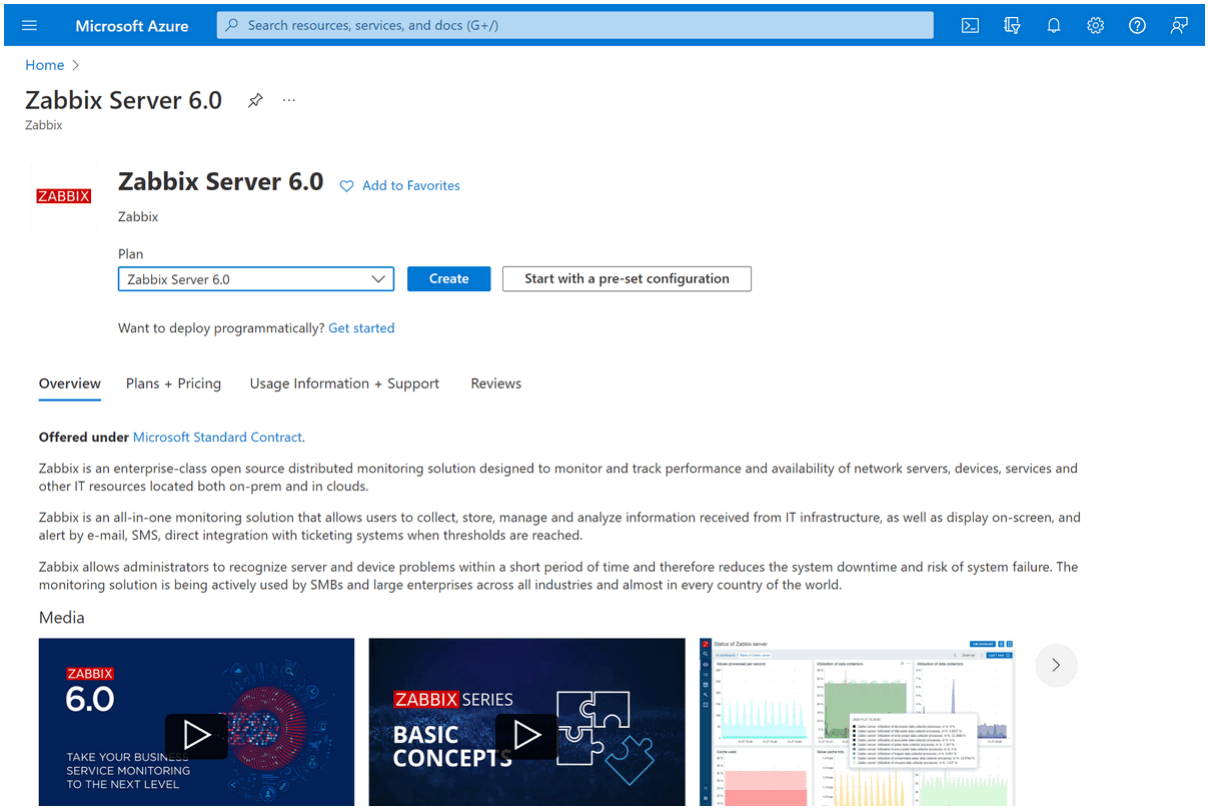
> **Attention:**
> Na platformie dostępnych jest kilka wersji Zabbix do wdrożenia. Upewnij się, że czytasz dokumentację dotyczącą tej samej wersji Zabbix, którą wdrożono. Dokumentację dotyczącą różnych wersji Zabbix można wybrać z menu rozwijanego wersji.

Wdrażanie appliance

Aby wdrożyć appliance Zabbix, najpierw przejdź do strony Zabbix w Azure Marketplace. Zapoznaj się z opisem, a następnie kliknij *Get it now*.



Zapoznaj się z omówieniem wdrożenia, planami cenowymi i innymi szczegółami. Następnie kliknij *Create* lub *Start with a pre-set configuration*, aby wdrożyć appliance Zabbix.

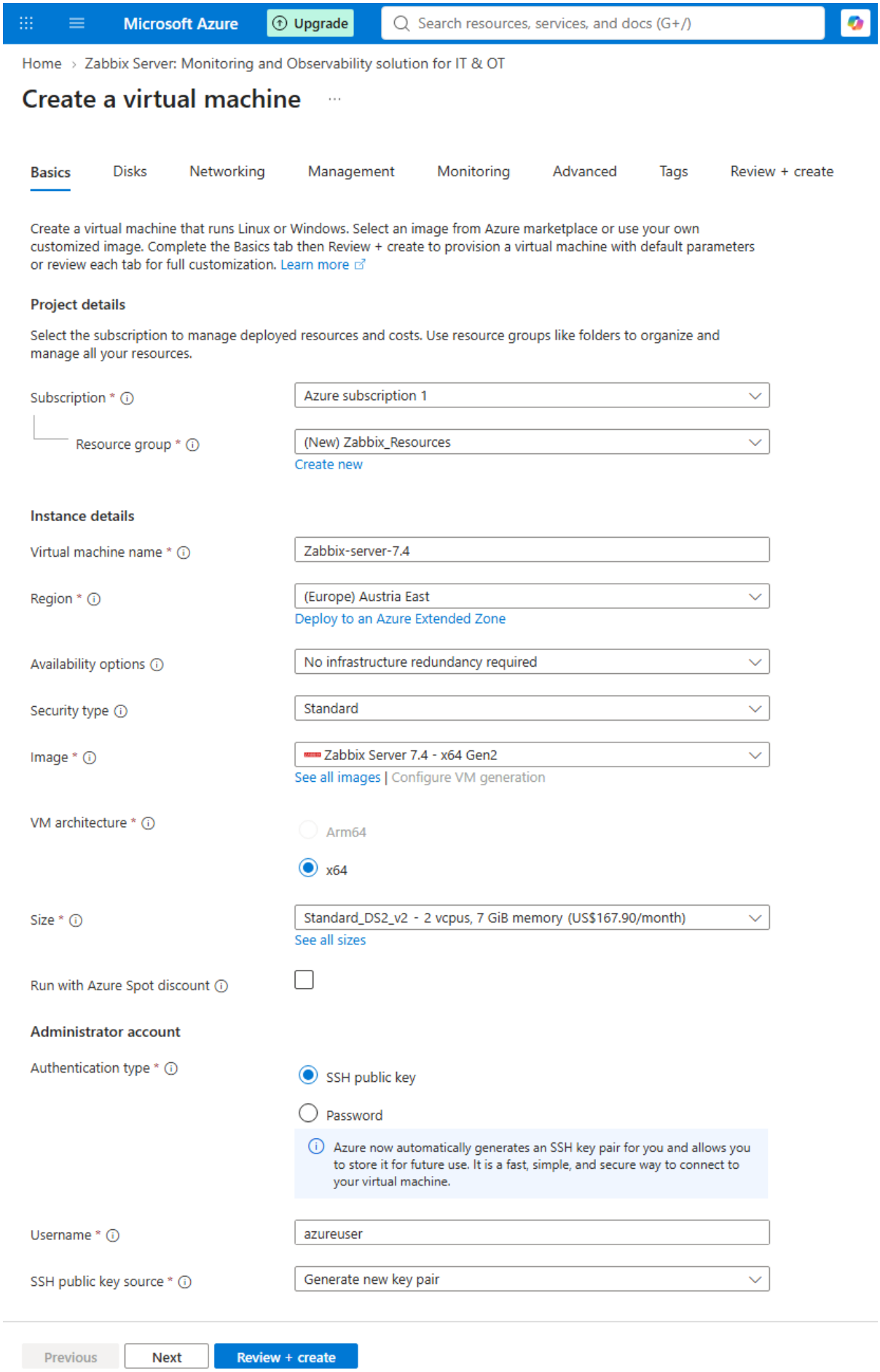


Skonfiguruj maszynę wirtualną, wybierając jej rozmiar, tworząc konto administratora oraz definiując ustawienia dysku, sieci, zarządzania i inne. Ustawienia te mogą się różnić w zależności od wymagań infrastruktury.

Możesz użyć poniższej konfiguracji jako przykładu:

Karta *Basics*:

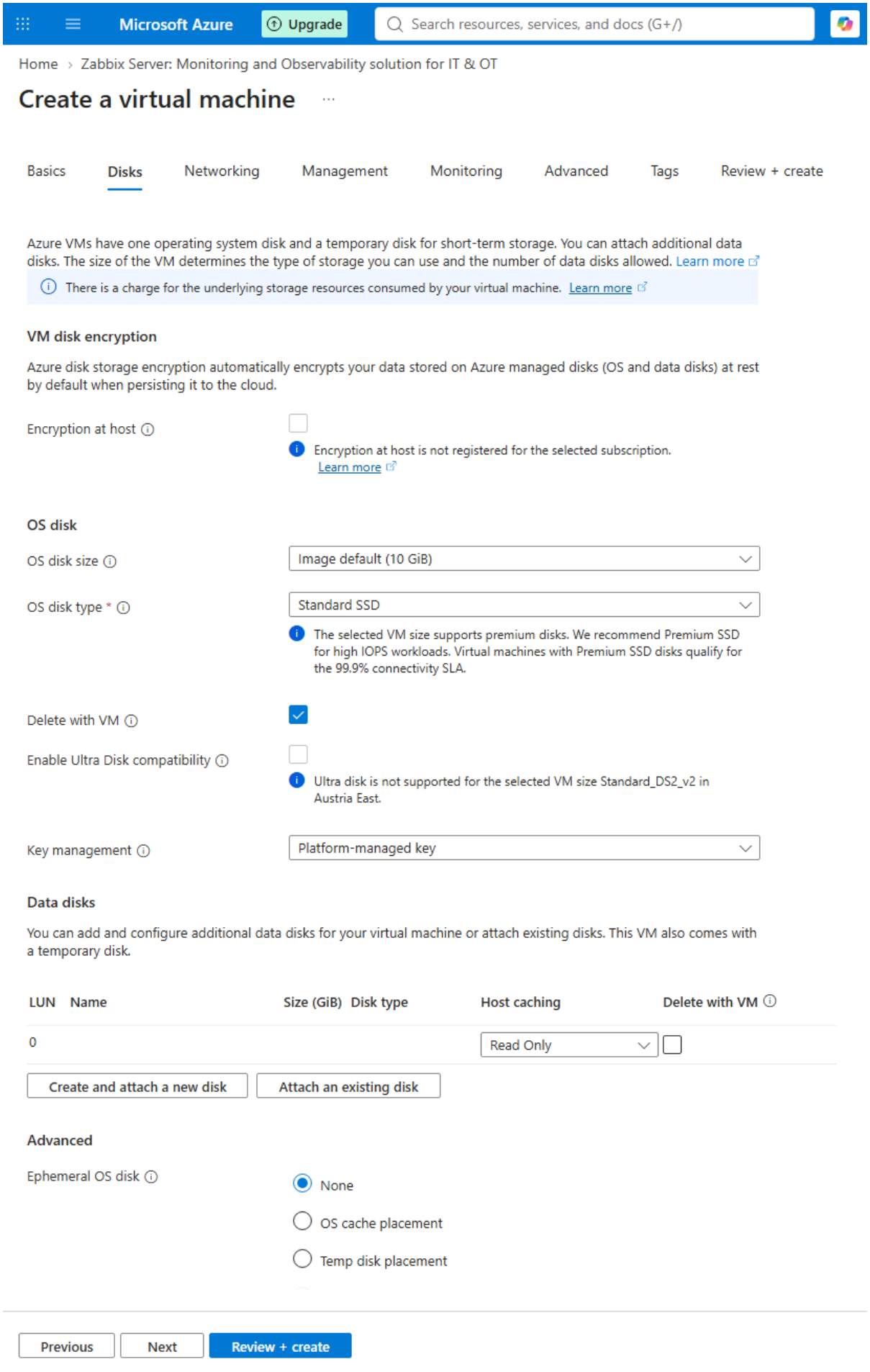
- *Subscription*: Azure subscription 1
- *Resource group*: Zabbix_Resources
- *Virtual machine name*: Zabbix-server--7.4
- *Availability options*: No infrastructure redundancy required
- *Security type*: Standard
- *Image*: Zabbix Server 7.4 - x64 Gen2
- *Size*: Standard_DS2_V2 - 2 vcpus, 7 GiB memory
- *Authentication type*: SSH public key
- *Username*: azureuser
- *SSH public key source*: Generate new key pair



Karta *Disks*:

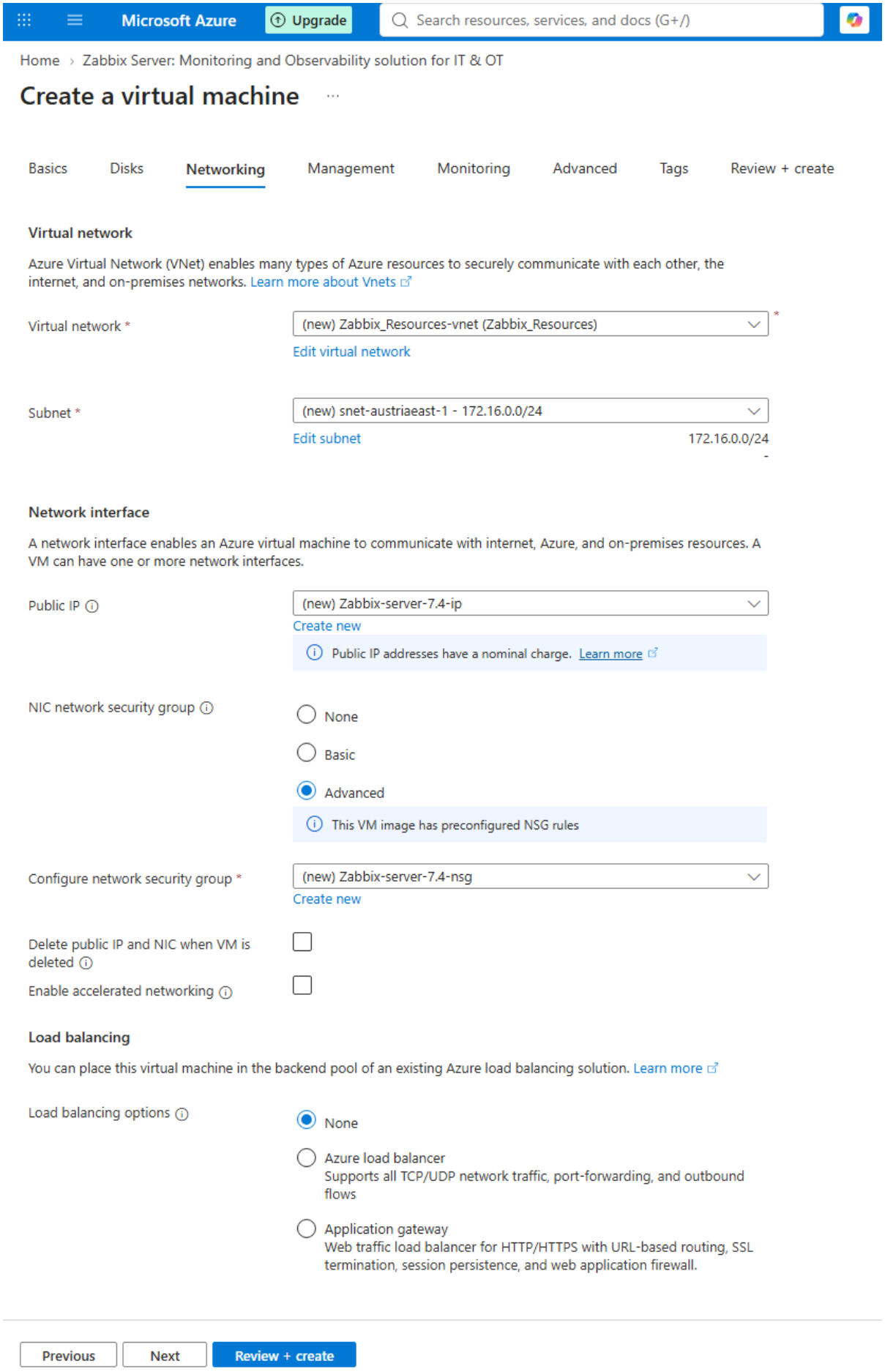
- *Encryption at host*: unchecked
- *OS disk type*: Standard SSD

- *Delete with VM*: checked
- *Enable Ultra Disk compatibility*: unchecked

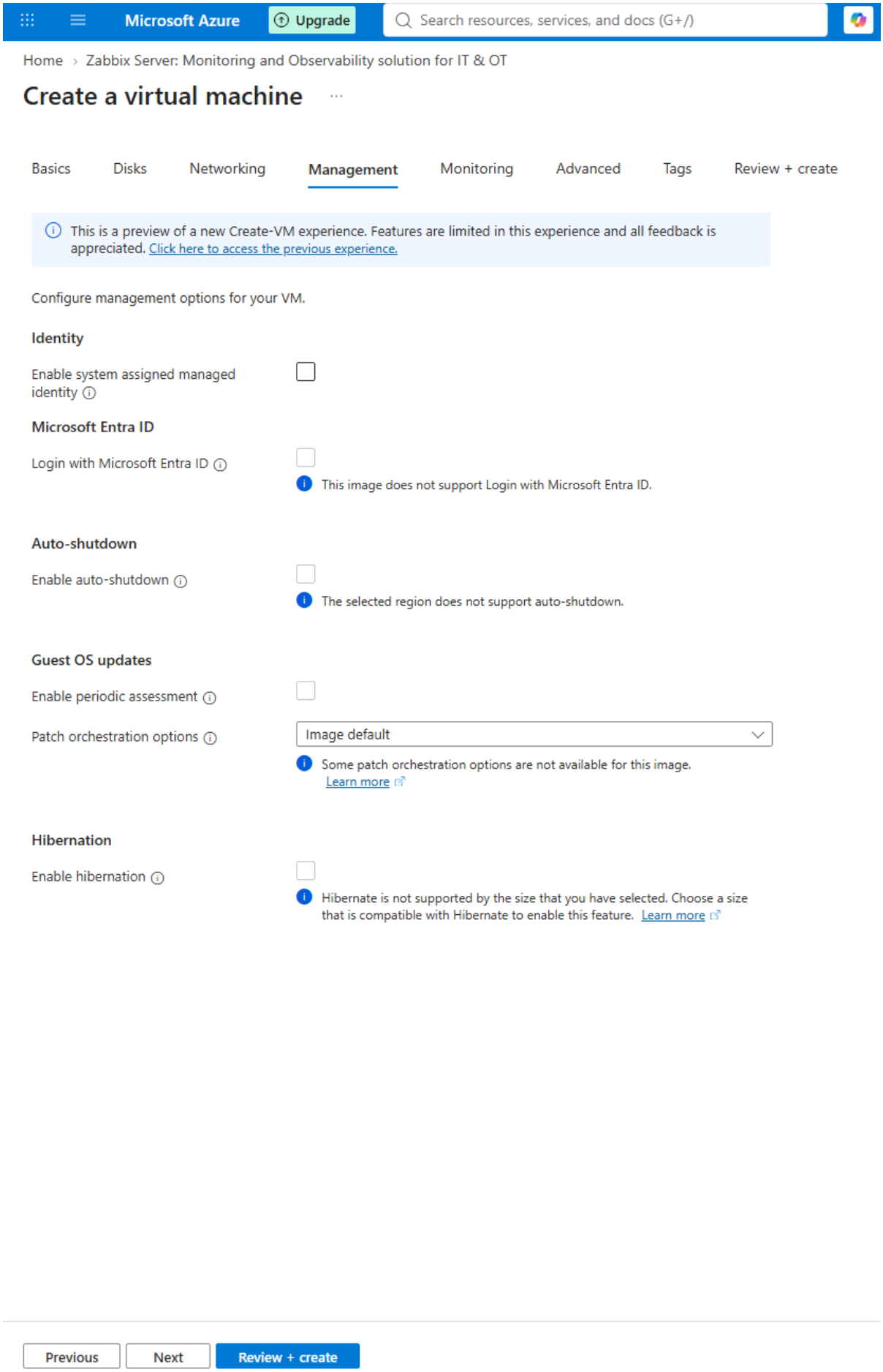


Karta *Networking*:

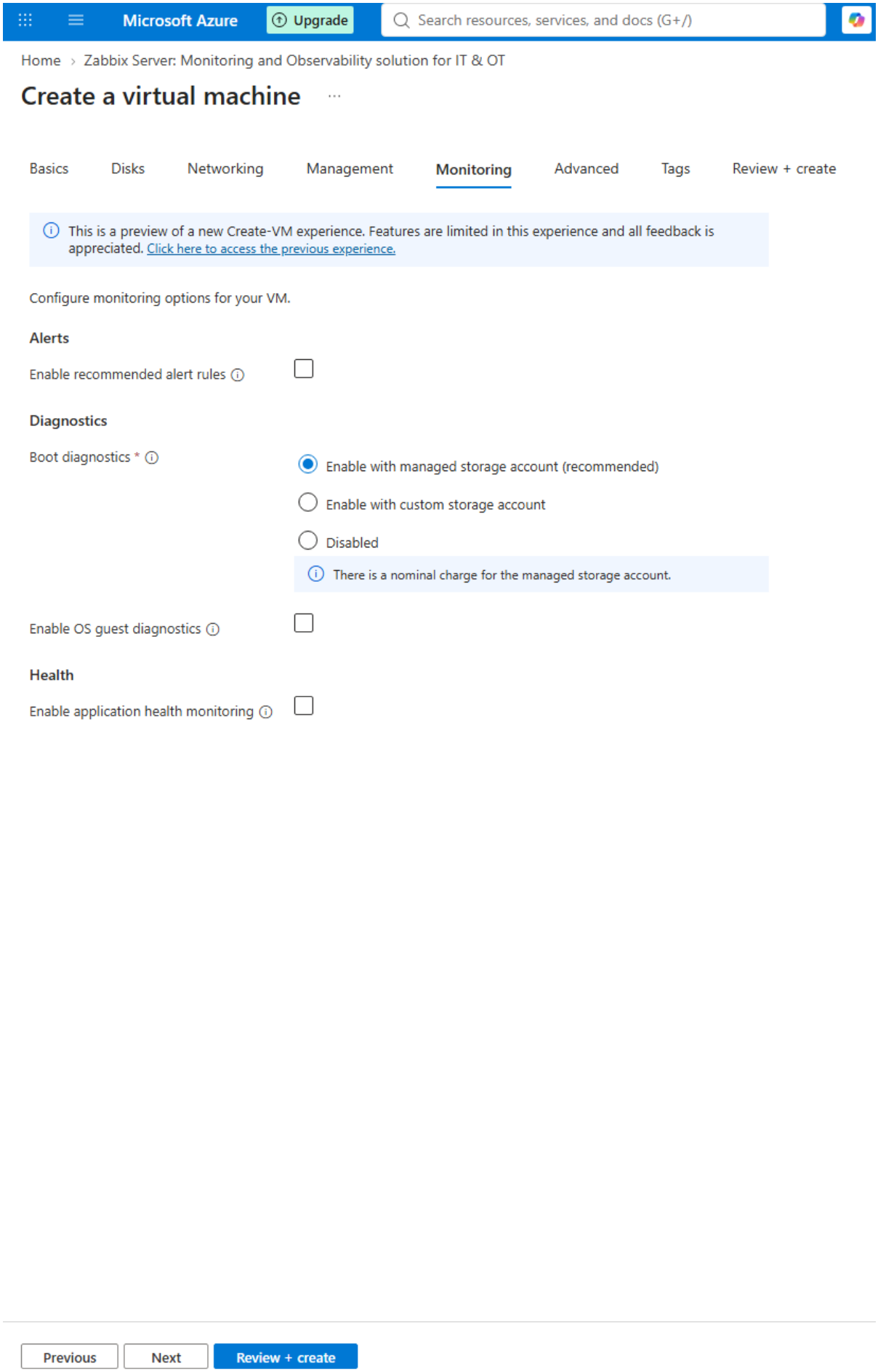
- *Virtual network*: Zabbix_Resources-vnet
- *Public IP*: Zabbix-server-7.4-ip
- *NIC network security group*: Advanced
- *Configure network security group*: Zabbix-server-7.4-nsg



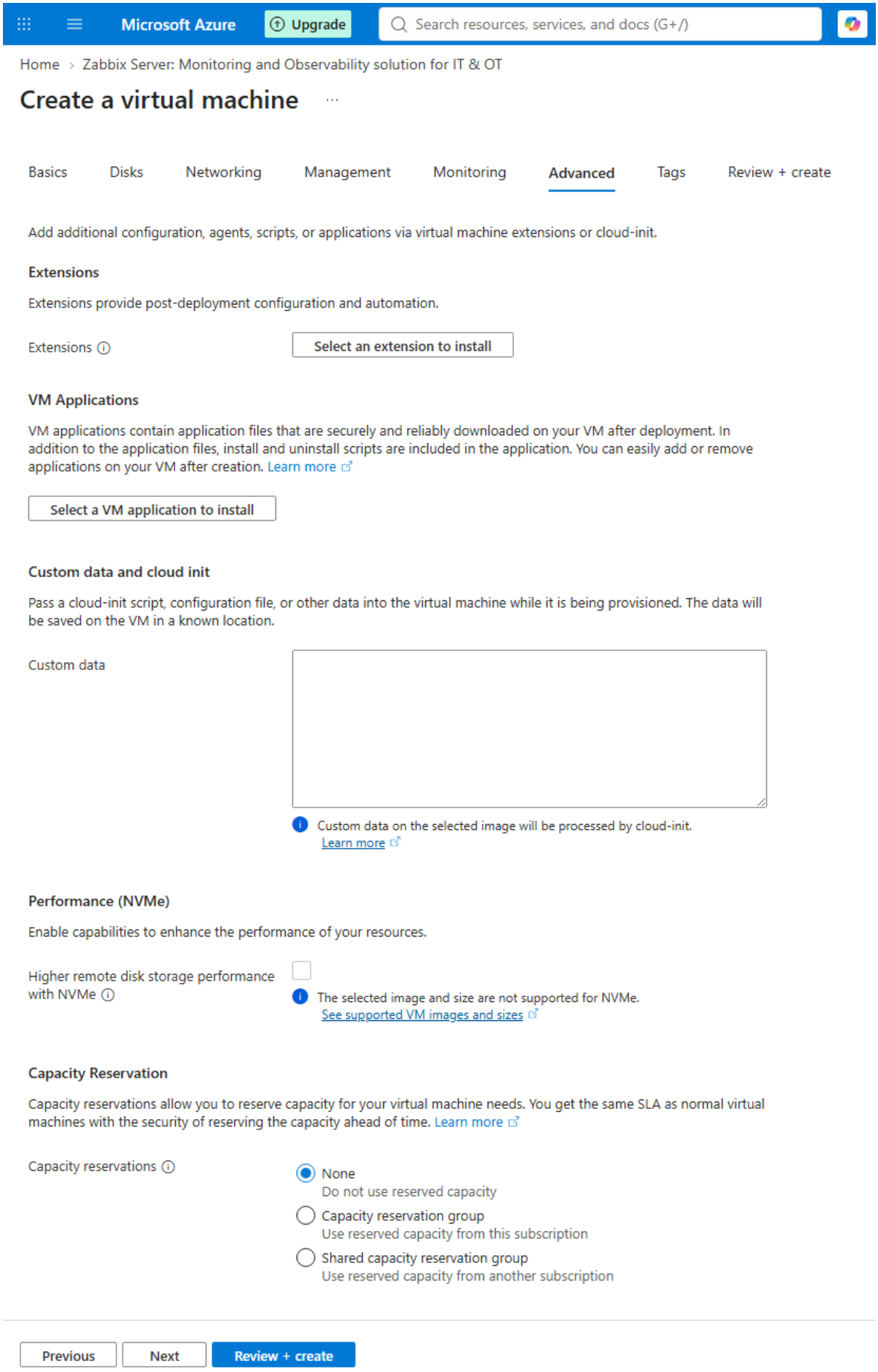
Karta *Management*: parametry można pozostawić domyślne.



Karta *Monitoring*: parametry można pozostawić domyślne.



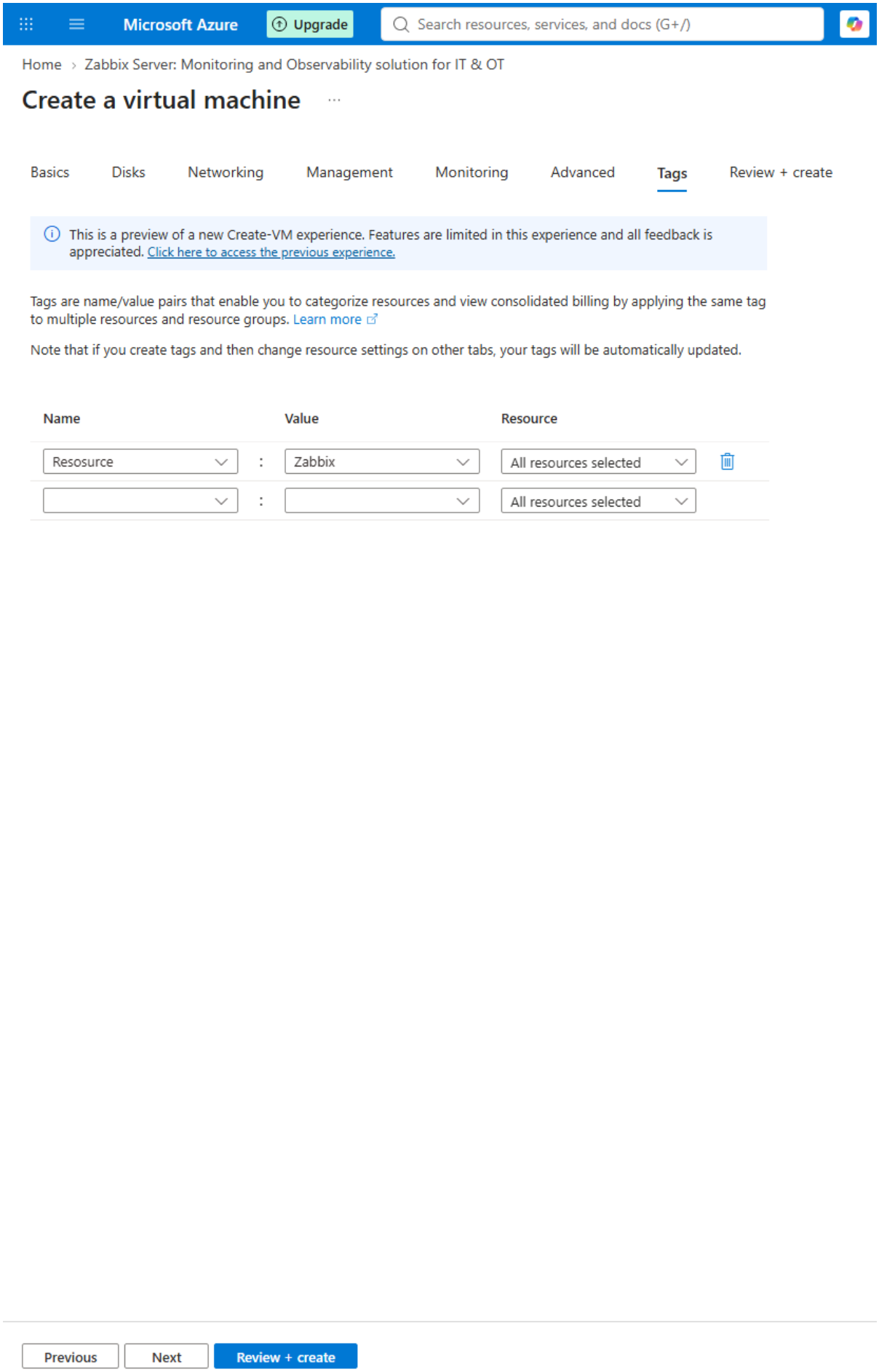
Karta *Advanced*: parametry można pozostawić domyślne.



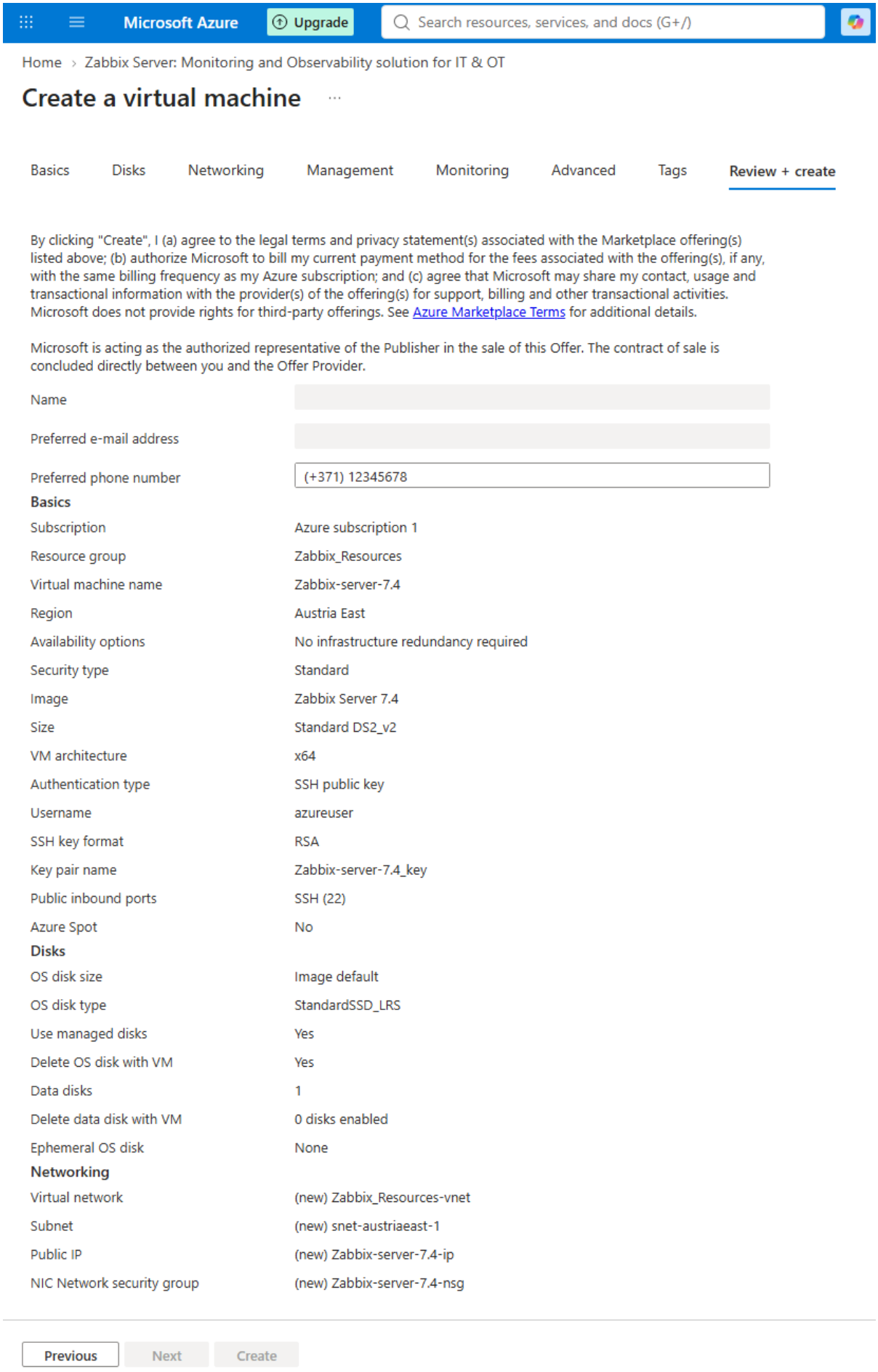
Karta *Tags*:

- *Name*: Resource
- *Value*: Zabbix

- *Resource*: All resource selected



Potwierdź szczegóły maszyny wirtualnej i utwórz ją.



Zostaniesz przekierowany na stronę stanu wdrożenia maszyny wirtualnej. Poczekaj na zakończenie wdrażania.

Po zakończeniu wdrażania otwórz zasób i zanotuj jego publiczny adres IP. Teraz musisz połączyć się z serwerem Zabbix za pomocą SSH (lub innej dostępnej metody), aby uzyskać dane logowania do frontendu Zabbix.

```
********************************************************************************

Zabbix frontend credentials:

Username: Admin

Password: zfpvagcs0s+srUUK+EM

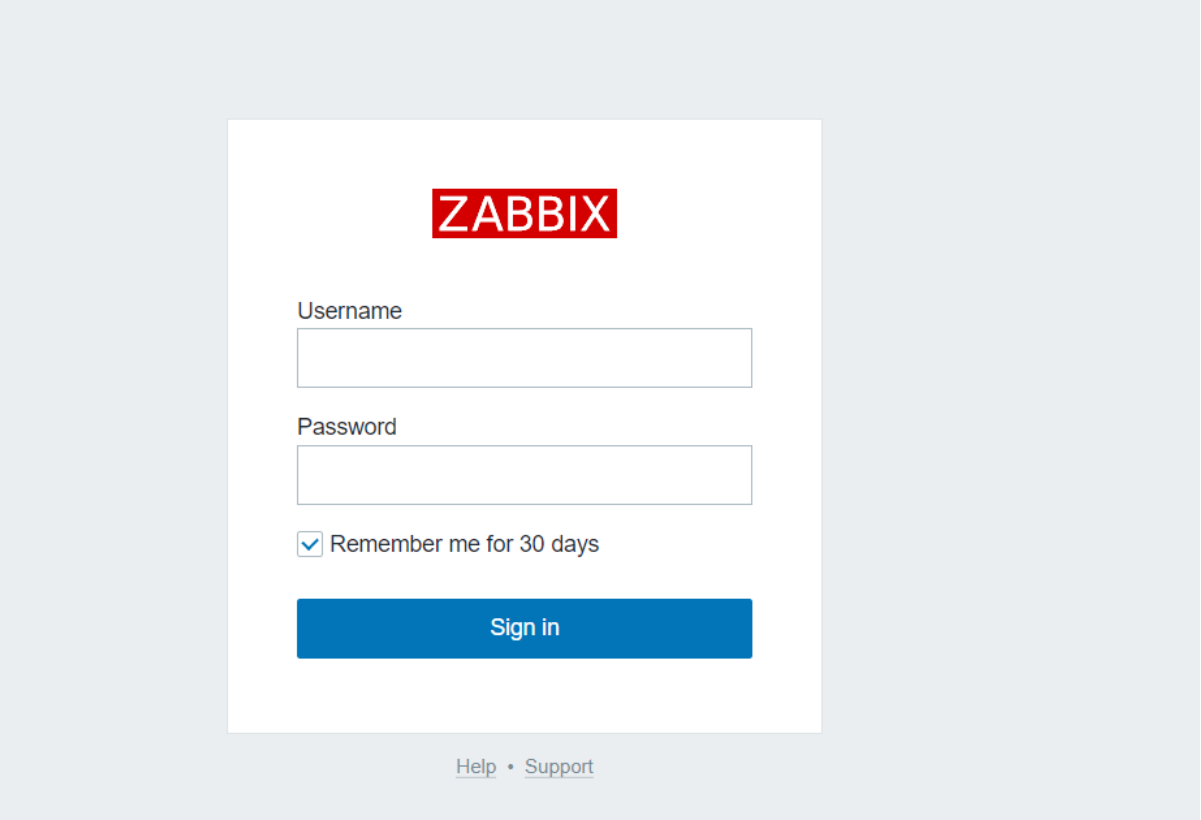

To learn about available professional services, including technical suppport and
 training, please visit https://www.zabbix.com/services

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 ~]$
```

Uzyskaj dostęp do frontendu Zabbix, otwierając publiczny adres IP appliance Zabbix w przeglądarce internetowej. Aby się zalogować, użyj nazwy użytkownika i hasła uzyskanych w poprzednim kroku.



Uzyskiwanie dostępu do bazy danych

Możesz uzyskać dostęp do bazy danych Zabbix za pomocą użytkownika `root` bez podawania hasła. Dane uwierzytelniające są przechowywane w pliku konfiguracyjnym `/root/.my.cnf`.

Przewodnik wdrażania proxy Zabbix na platformie Azure

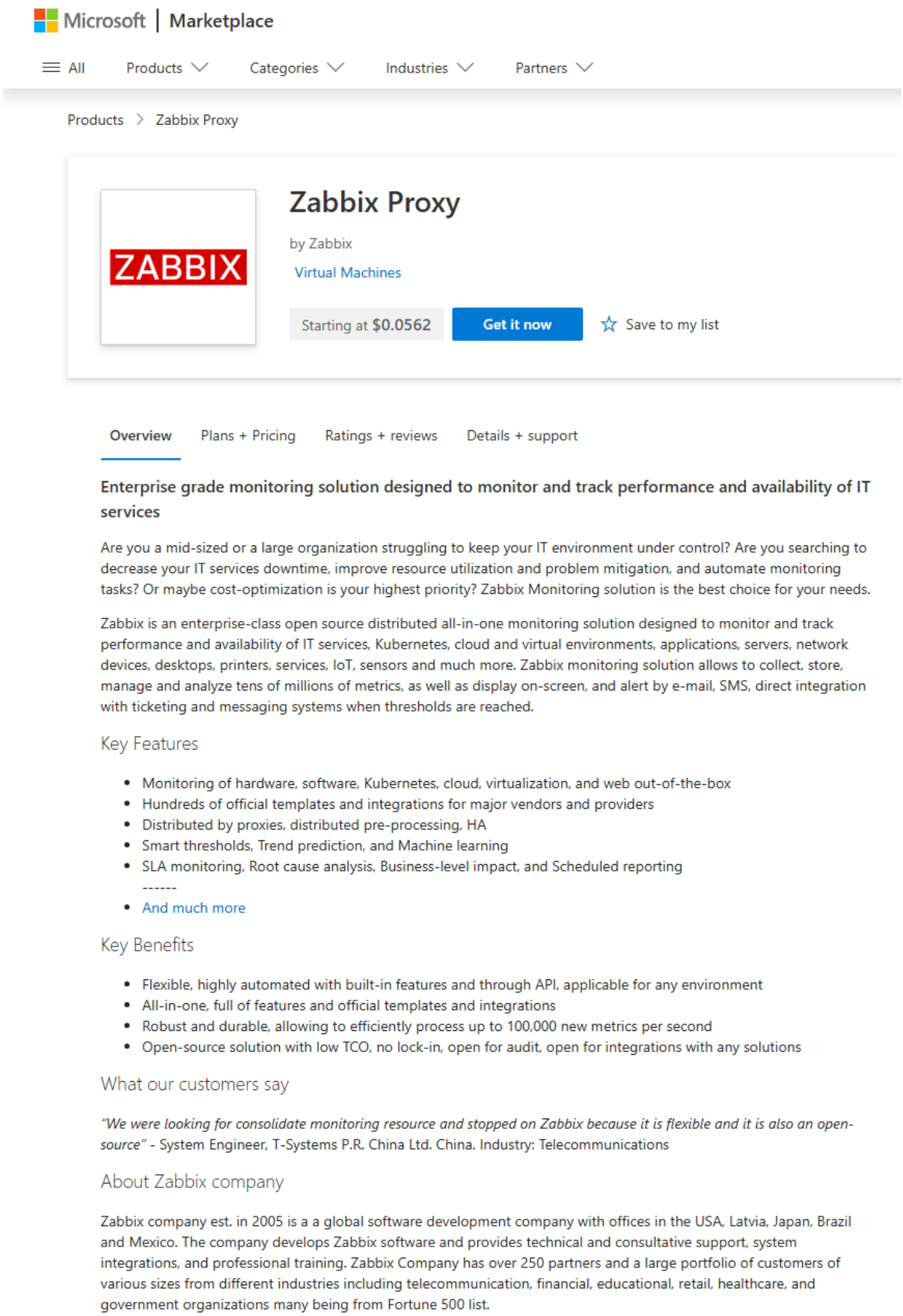
Przegląd

Ta strona opisuje sposób wdrażania proxy Zabbix przy użyciu platformy Azure.

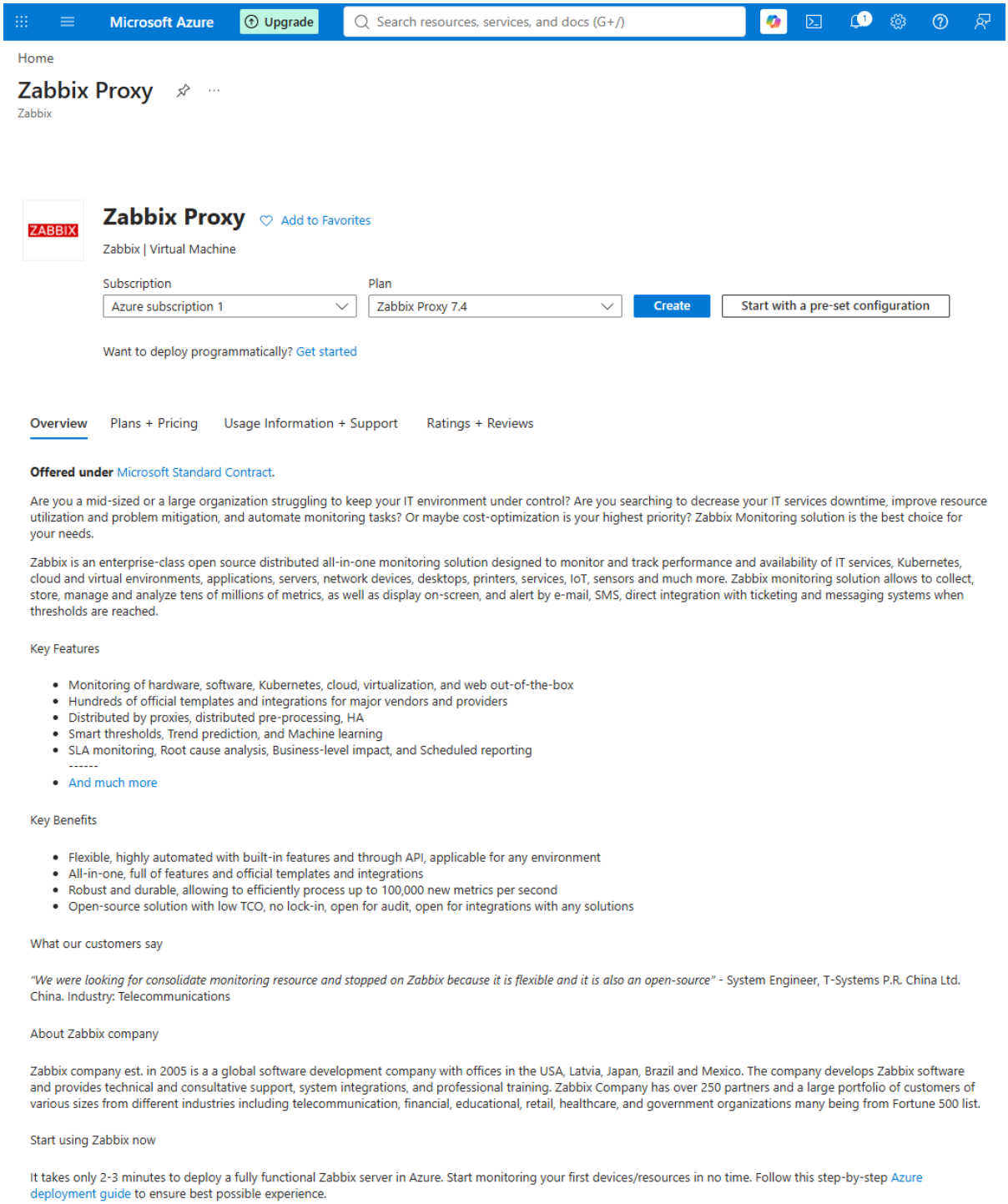
Attention:
Na platformie dostępnych jest kilka wersji Zabbix do wdrożenia. Upewnij się, że czytasz dokumentację dotyczącą tej samej wersji Zabbix, którą wdrożono. Dokumentację dotyczącą różnych wersji Zabbix można wybrać z menu rozwijanego wersji.

Wdrażanie proxy Zabbix

Aby wdrożyć proxy Zabbix, najpierw przejdź do strony Zabbix w witrynie Azure Marketplace. Kliknij *Get it now*.



W tym miejscu możesz przeczytać dodatkowe informacje dotyczące bieżącego wdrożenia, takie jak ogólny przegląd, plany cenowe i inne. Kliknij *Create* lub *Start with a pre-set configuration*.

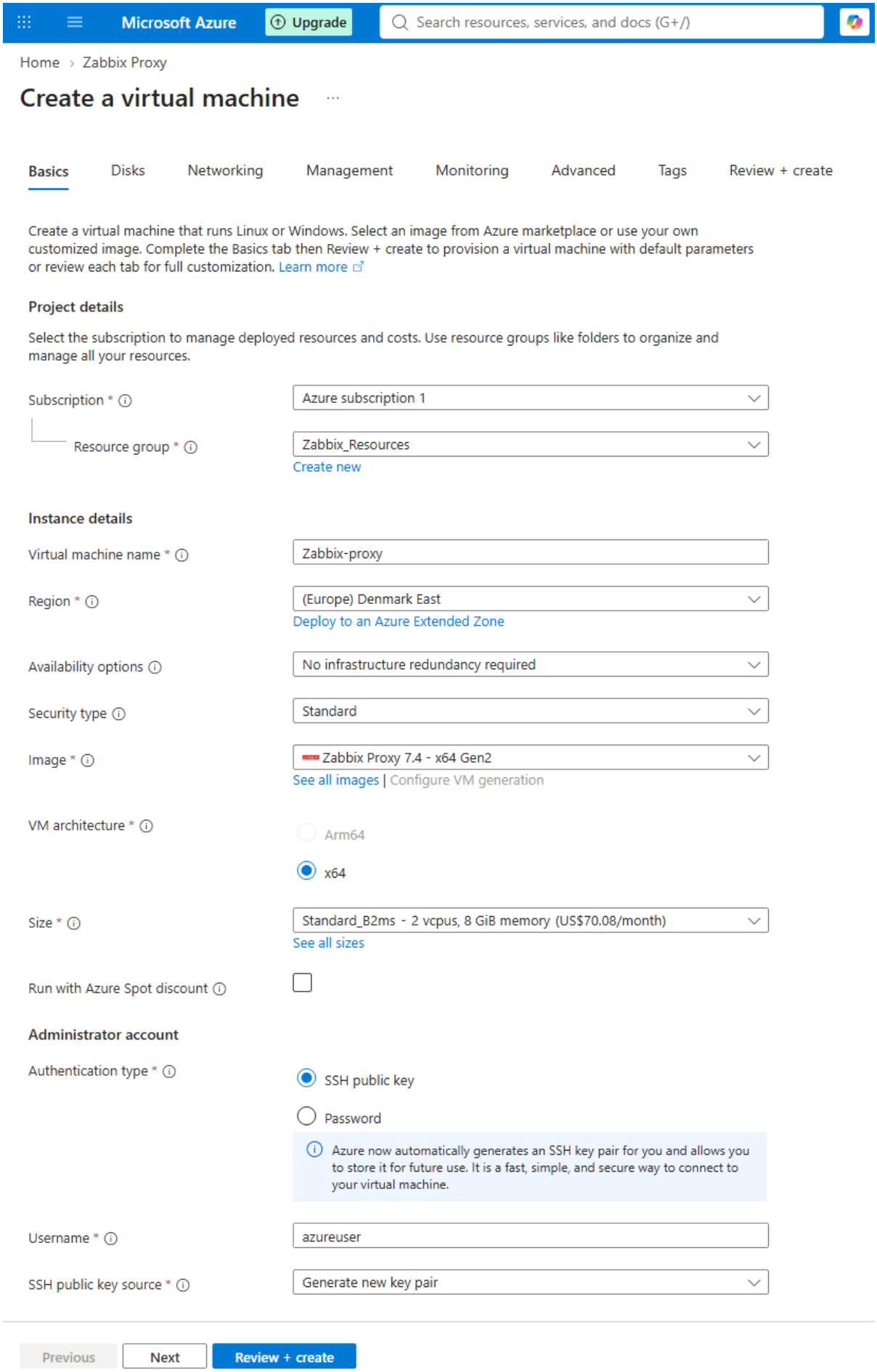


Przejdź do szczegółowej konfiguracji maszyny wirtualnej (VM): wybierz rozmiar VM, konto administratora, dysk, sieć, zarządzanie i inne ustawienia. Ustawienia te mogą się różnić w zależności od wymagań infrastruktury.

Możesz użyć poniższej konfiguracji jako przykładu:

Karta *Basics*:

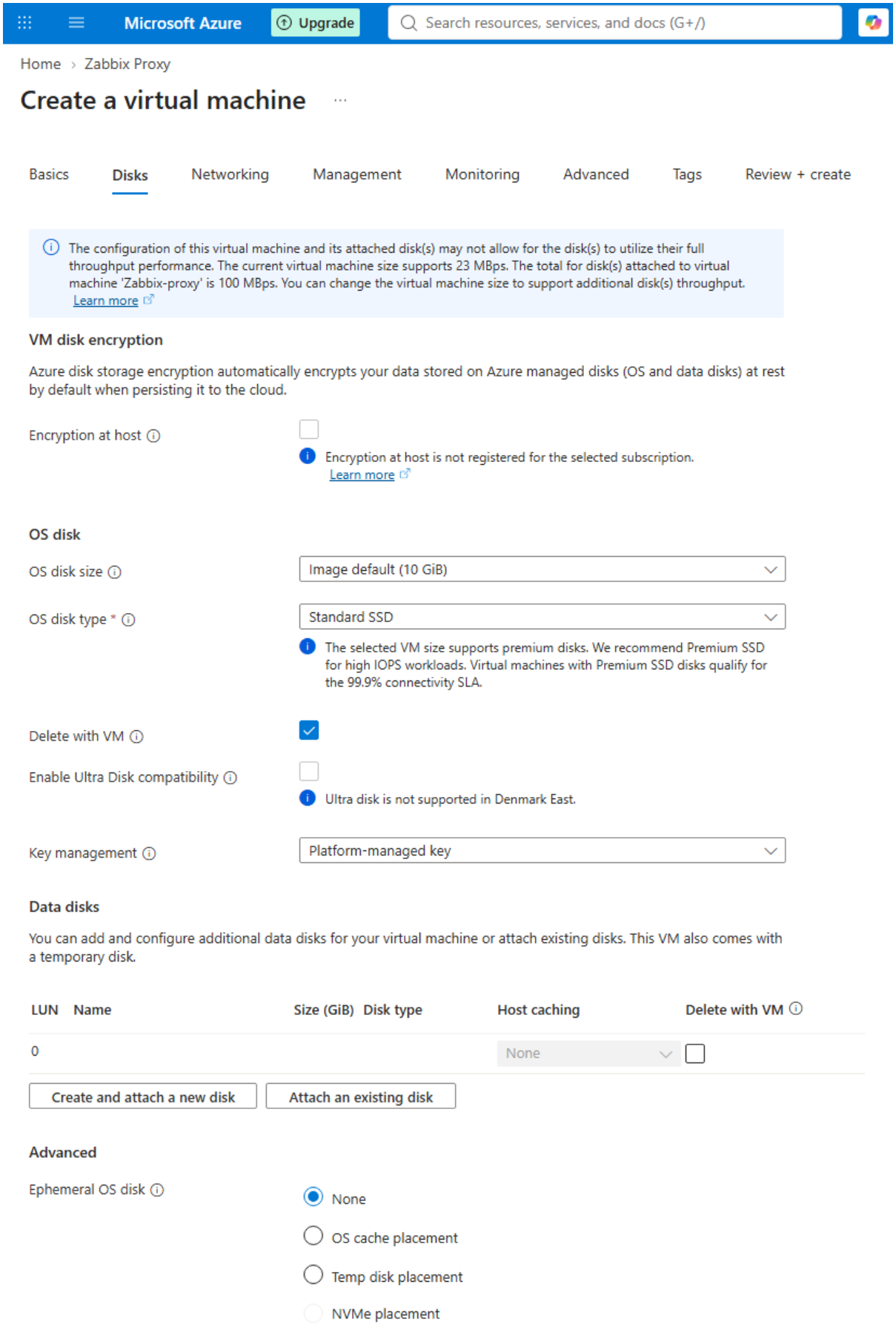
- *Subscription*: Azure subscription 1
- *Resource group*: Zabbix_Resources
- *Virtual machine name*: Zabbix-proxy
- *Availability options*: No infrastructure redundancy required
- *Security type*: Standard
- *Image*: Zabbix Proxy 7.4 - x64 Gen2
- *Size*: Standard_B2ms - 2 vcpus, 8 GiB memory
- *Authentication type*: SSH public key
- *Username*: azureuser
- *SSH public key source*: Generate new key pair



Karta *Disks*:

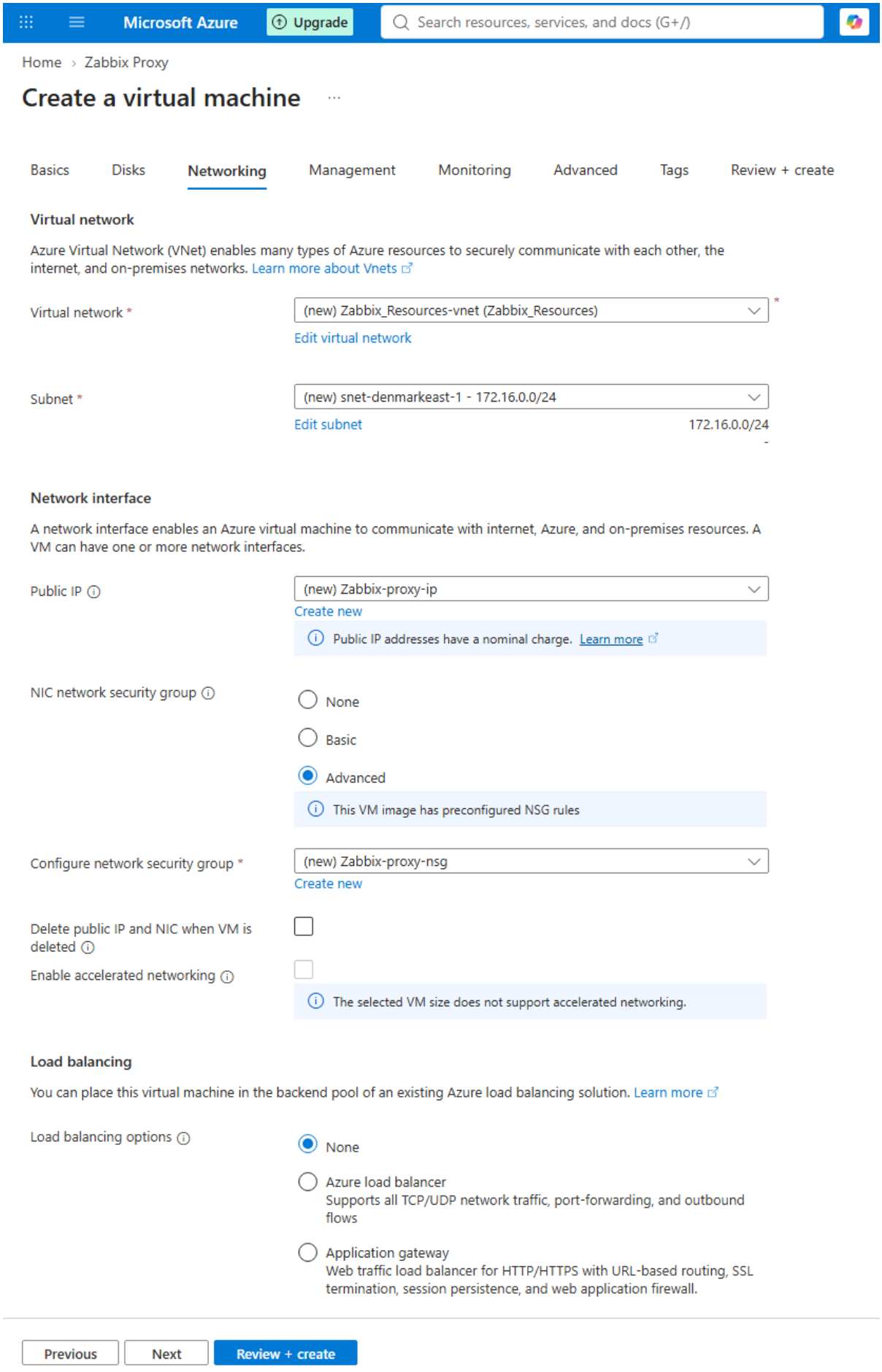
- *Encryption at host*: unchecked

- *OS disk type*: Standard SSD
- *Delete with VM*: checked
- *Enable Ultra Disk compatibility*: unchecked

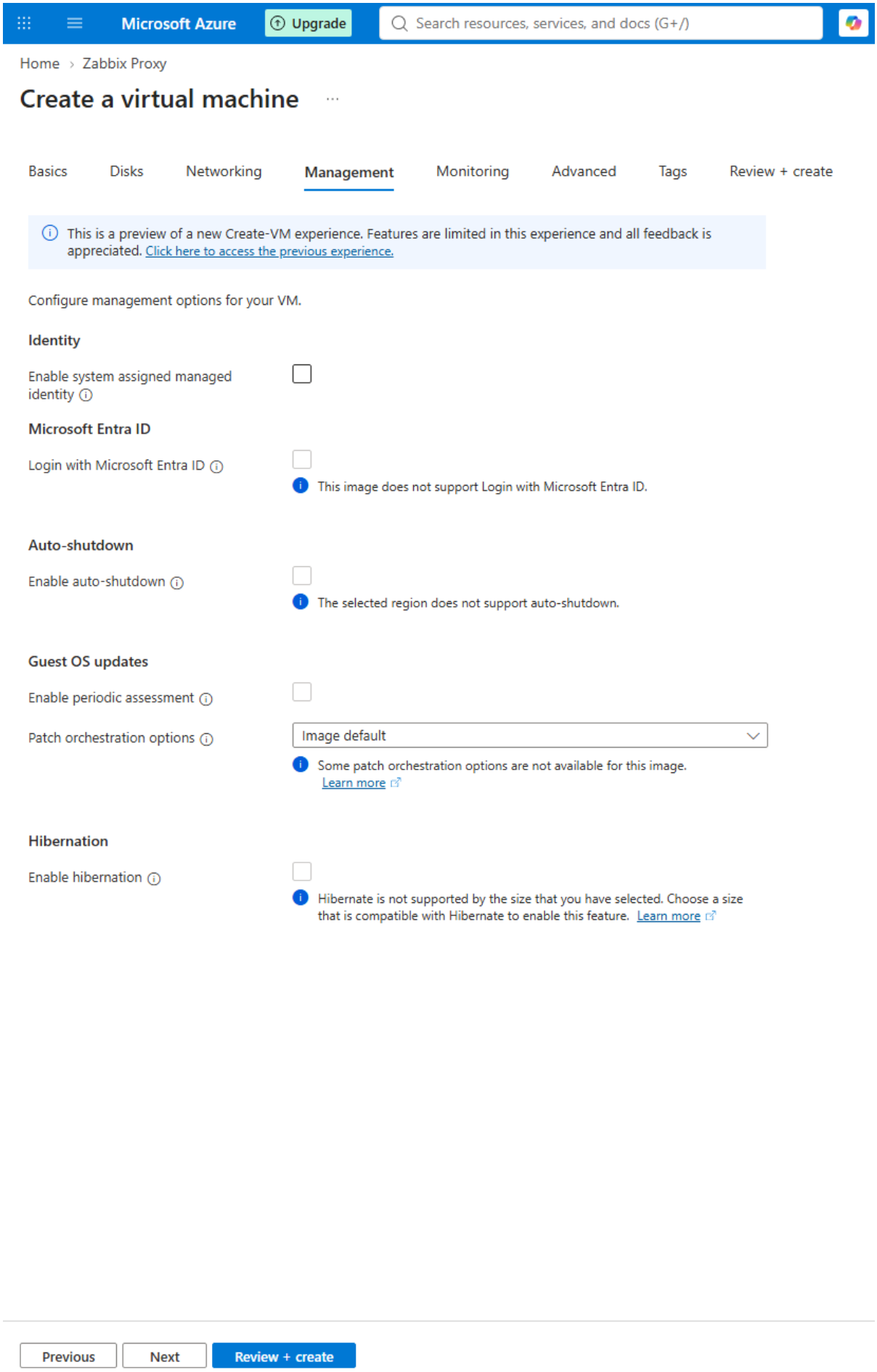


Karta *Networking*:

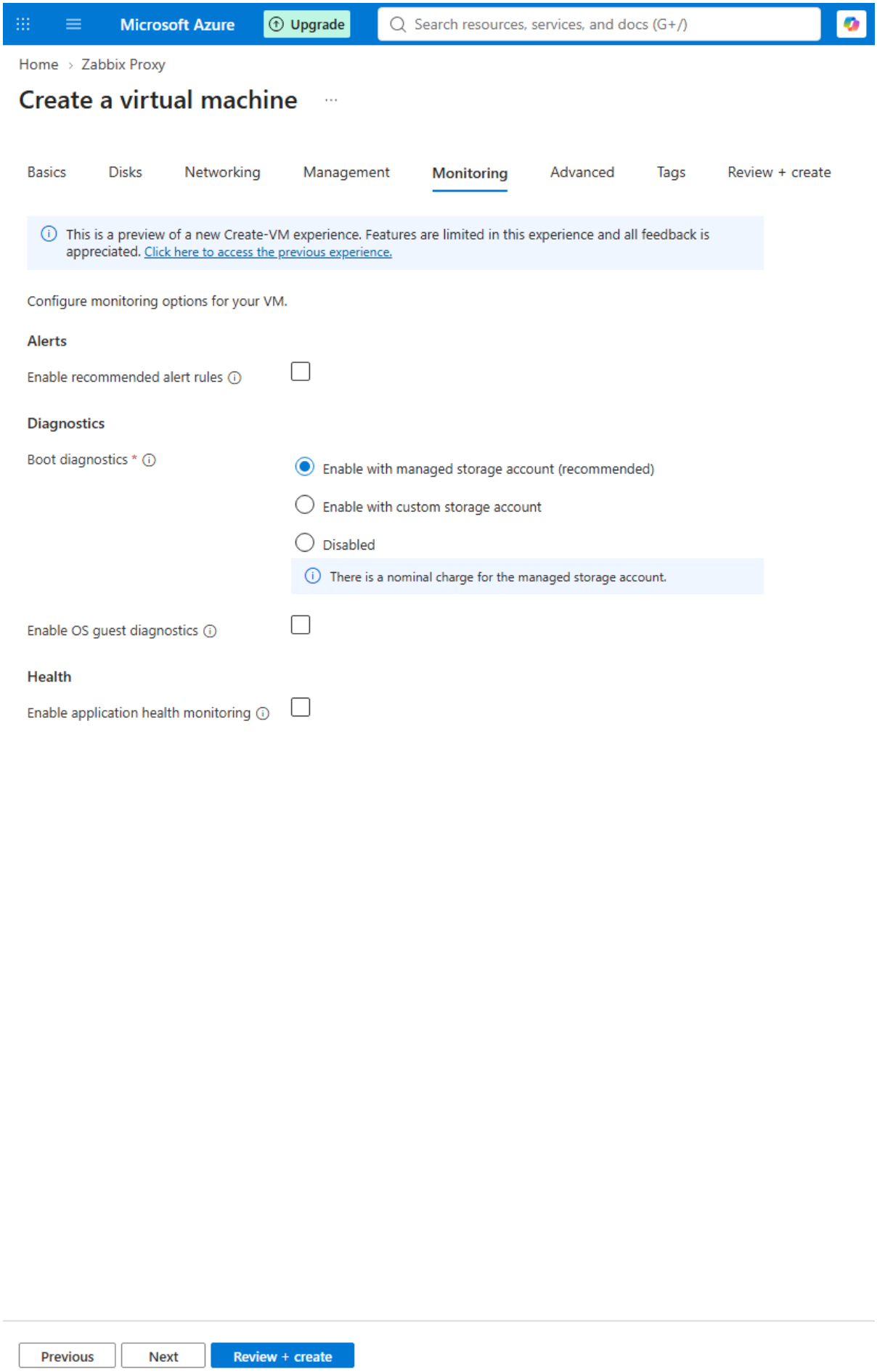
- *Virtual network*: Zabbix_Resources-vnet
- *Public IP*: Zabbix-proxy-ip
- *NIC network security group*: Advanced
- *Configure network security group*: Zabbix-proxy-nsg



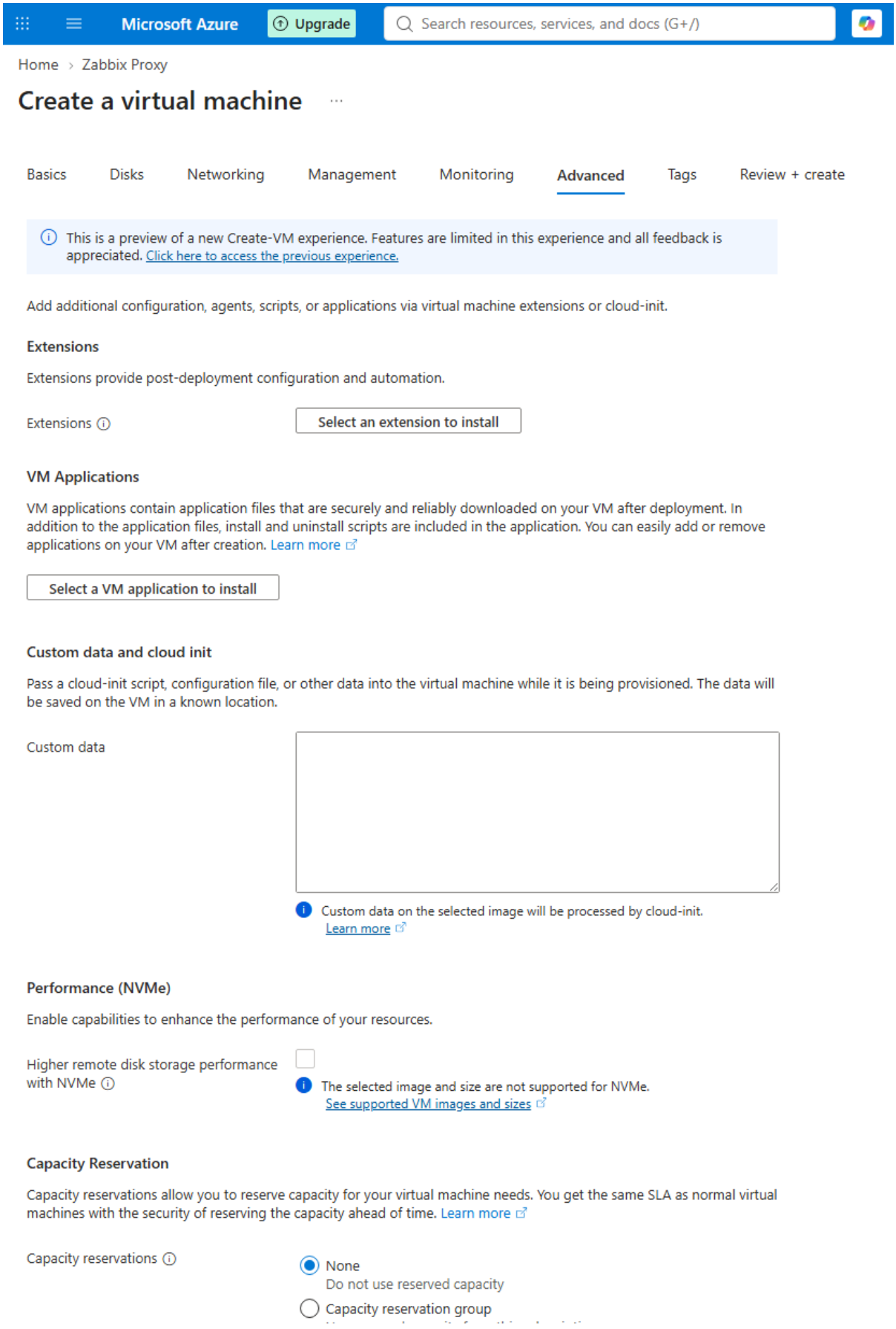
Karta *Management*: parametry można pozostawić bez zmian.



Karta *Monitoring*: parametry można pozostawić bez zmian.



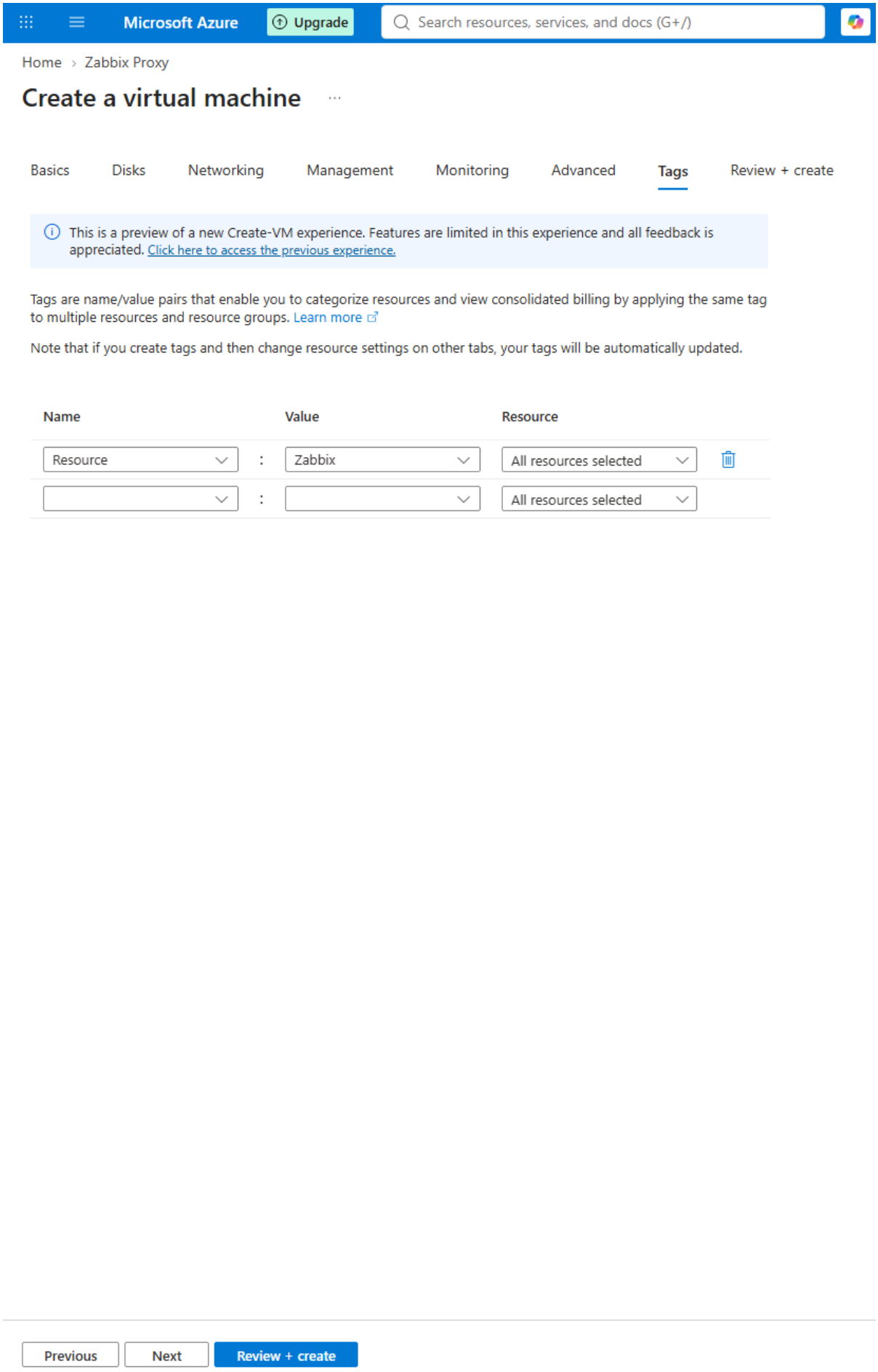
Karta *Advanced*: parametry można pozostawić bez zmian.



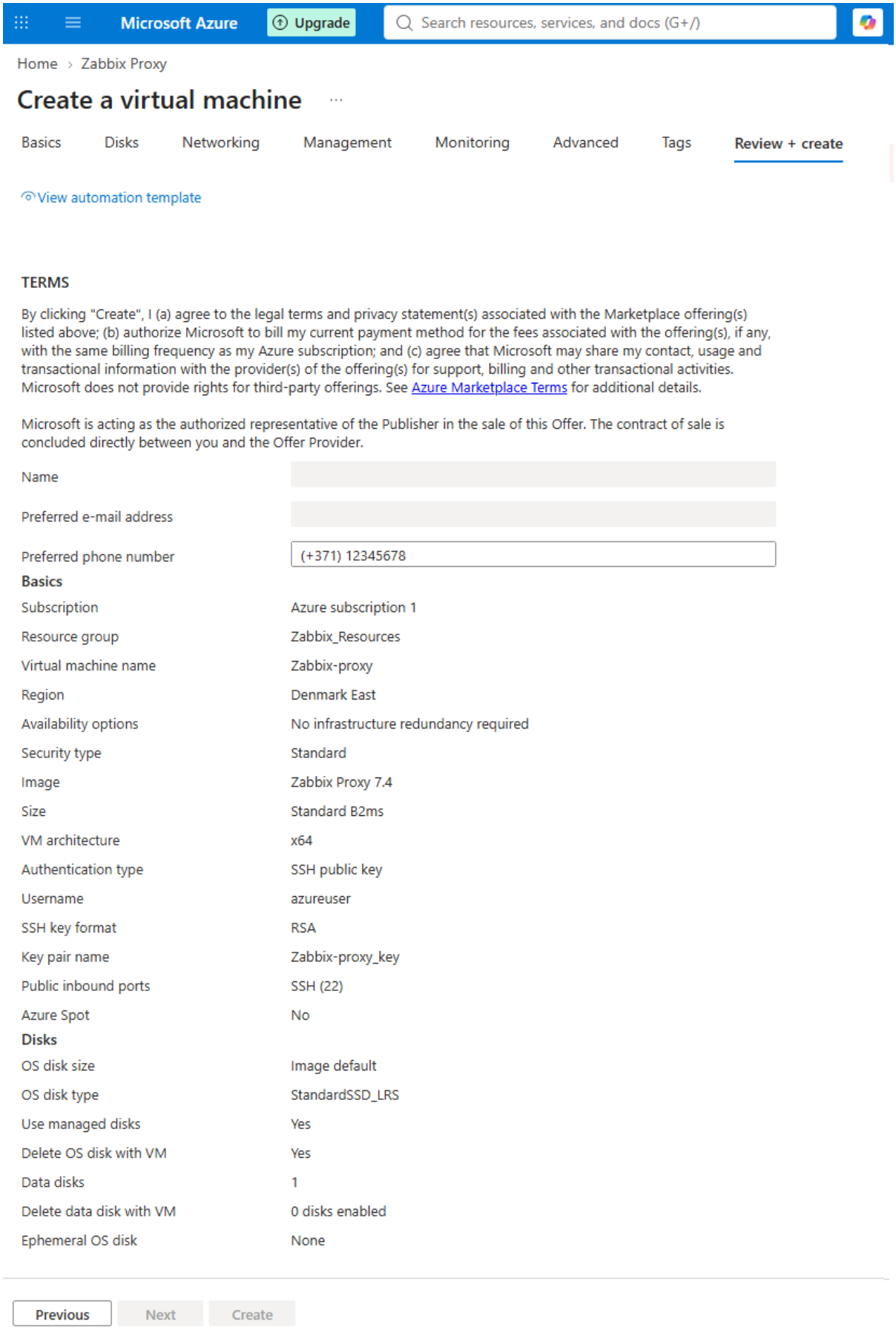
Karta *Tags*:

- *Name*: Resource
- *Value*: Zabbix

- *Resource*: All resource selected



Potwierdź szczegóły VM i utwórz VM.



Nastąpi przekierowanie na stronę stanu wdrożenia VM. Poczekaj na zakończenie wdrożenia.

Po zakończeniu wdrożenia otwórz zasób i zanotuj jego publiczny adres IP. Połącz się z proxy Zabbix za pomocą SSH i poświadczeń administratora utworzonych podczas wdrażania VM, aby skonfigurować 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 ~]$
```

Plik konfiguracyjny proxy Zabbix znajduje się w `/etc/zabbix/zabbix_proxy.conf`.

Początkowa konfiguracja proxy polega na zmodyfikowaniu parametrów `ProxyMode`, `Server` i `Hostname` w pliku konfiguracyjnym proxy Zabbix.

Tryb proxy można zmienić, edytując parametr `ProxyMode`. Ustaw `ProxyMode=0`, aby użyć trybu aktywnego, lub `ProxyMode=1`, aby użyć trybu pasywnego.

Parametr `Server` powinien zawierać adres IP serwera Zabbix.

Parametr `Hostname` powinien zawierać nazwę hosta proxy Zabbix. Zdefiniowana tutaj nazwa hosta powinna być zgodna z nazwą proxy podaną podczas konfiguracji proxy w frontend 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
```

Otwórz frontend Zabbix i przejdź do *Administration* > *Proxies*. Kliknij przycisk *Create proxy*.

Wprowadź tę samą nazwę proxy co wartość parametru *Hostname* w pliku konfiguracyjnym proxy.

Wybierz tryb proxy i podaj adres IP proxy (opcjonalnie w przypadku aktywnych proxy).

Kliknij przycisk *Create proxy*, aby utworzyć proxy.

Lokalizacja bazy danych proxy

Bazę danych SQLite proxy można znaleźć w `/tmp/zabbix_proxy.sqlite3`; jest ona określona przez parametr `DBName` w pliku konfiguracyjnym proxy Zabbix.

Przewodnik wdrażania w Google Cloud

Przegląd

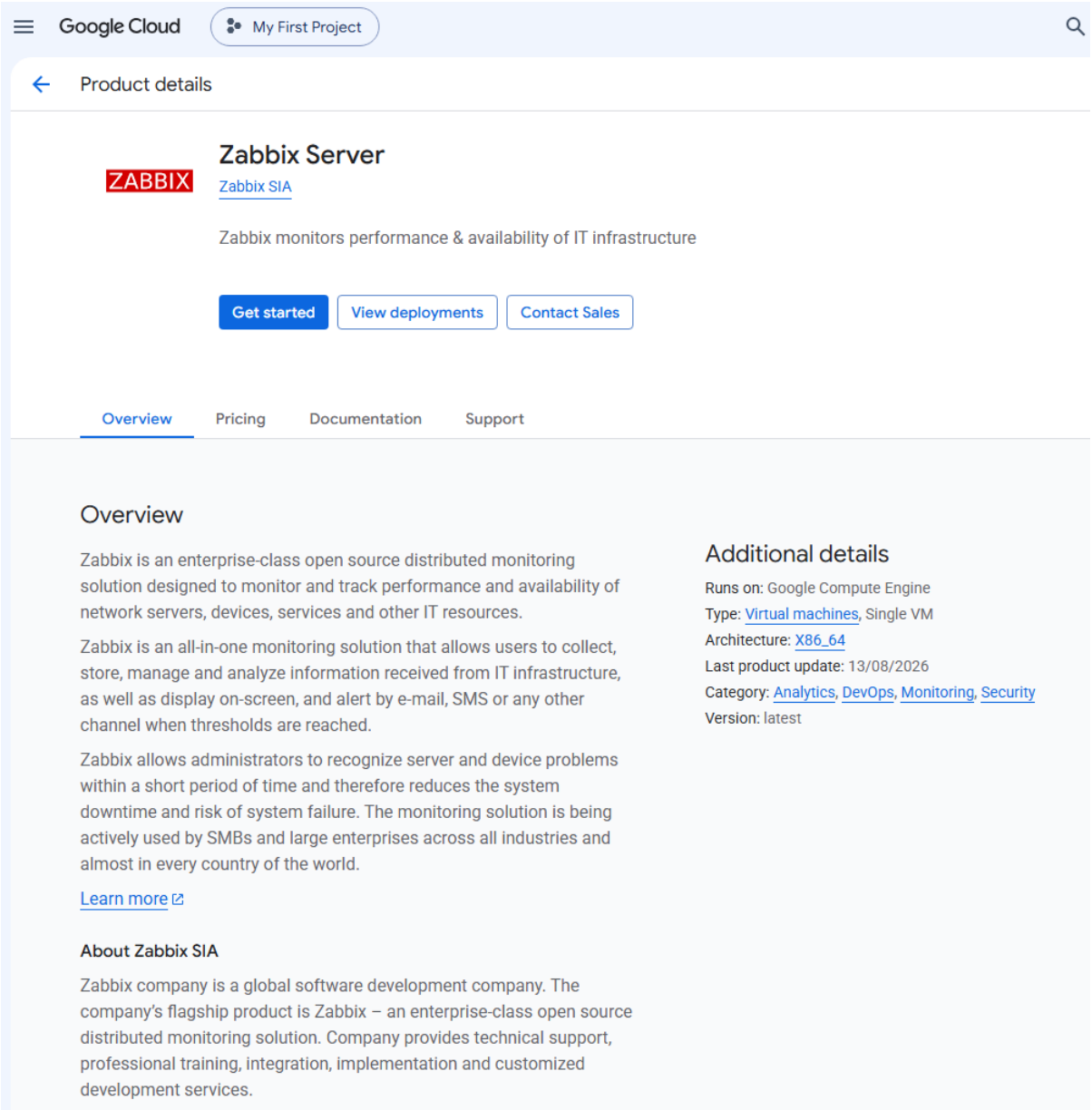
Ta strona opisuje sposób wdrażania Zabbix za pomocą Google Cloud Marketplace.

Attention:
Na platformie dostępnych jest kilka wersji Zabbix do wdrożenia. Upewnij się, że czytasz dokumentację dotyczącą tej samej wersji Zabbix, którą wdrożono. Dokumentację dotyczącą różnych wersji Zabbix można wybrać z menu rozwijanego wersji.

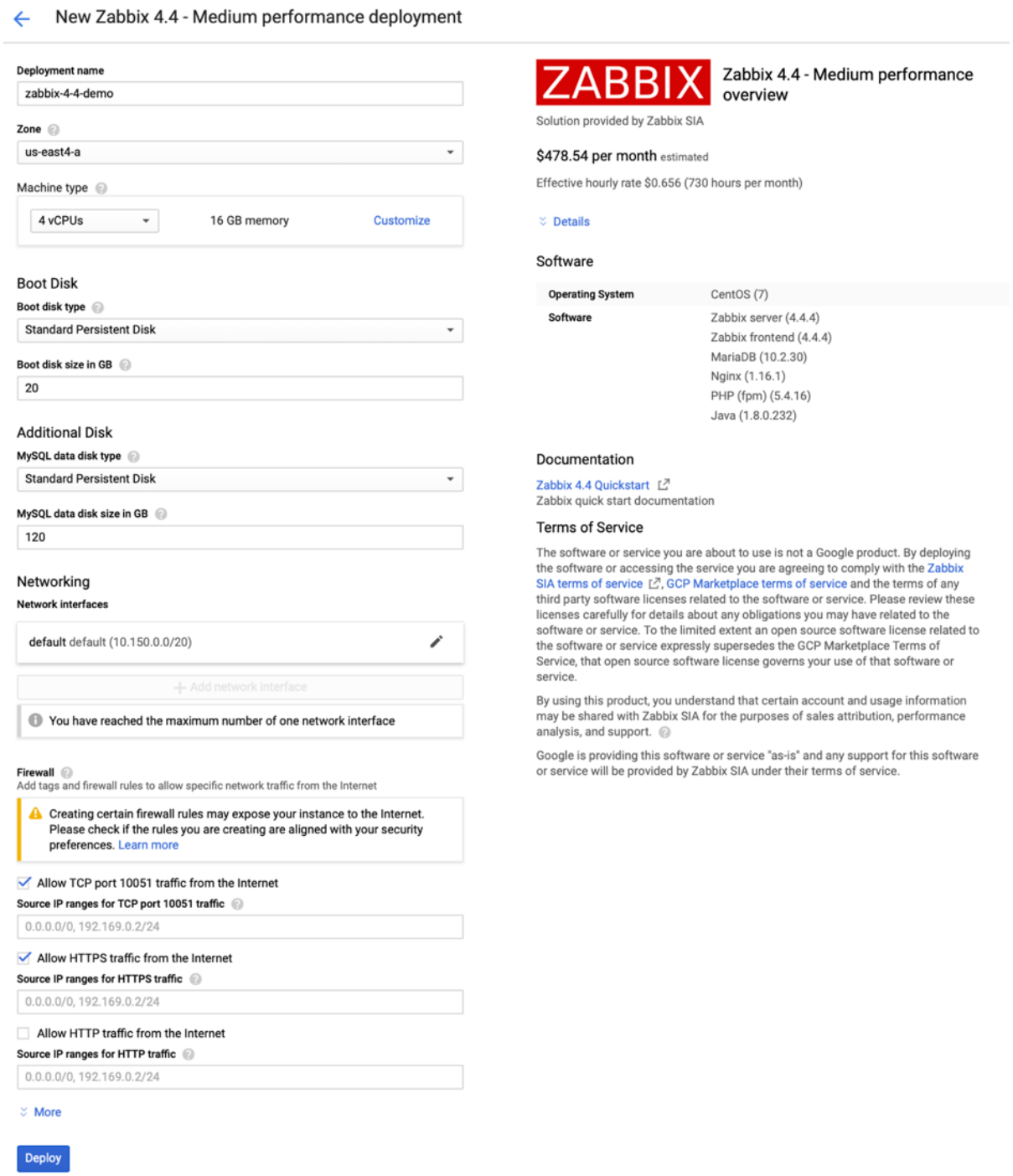
Wdrażanie appliance

Najpierw wdróż appliance Zabbix. Dzięki Google Cloud Marketplace jest to naprawdę proste. Wystarczy przejść do produktu Zabbix Server w Marketplace.

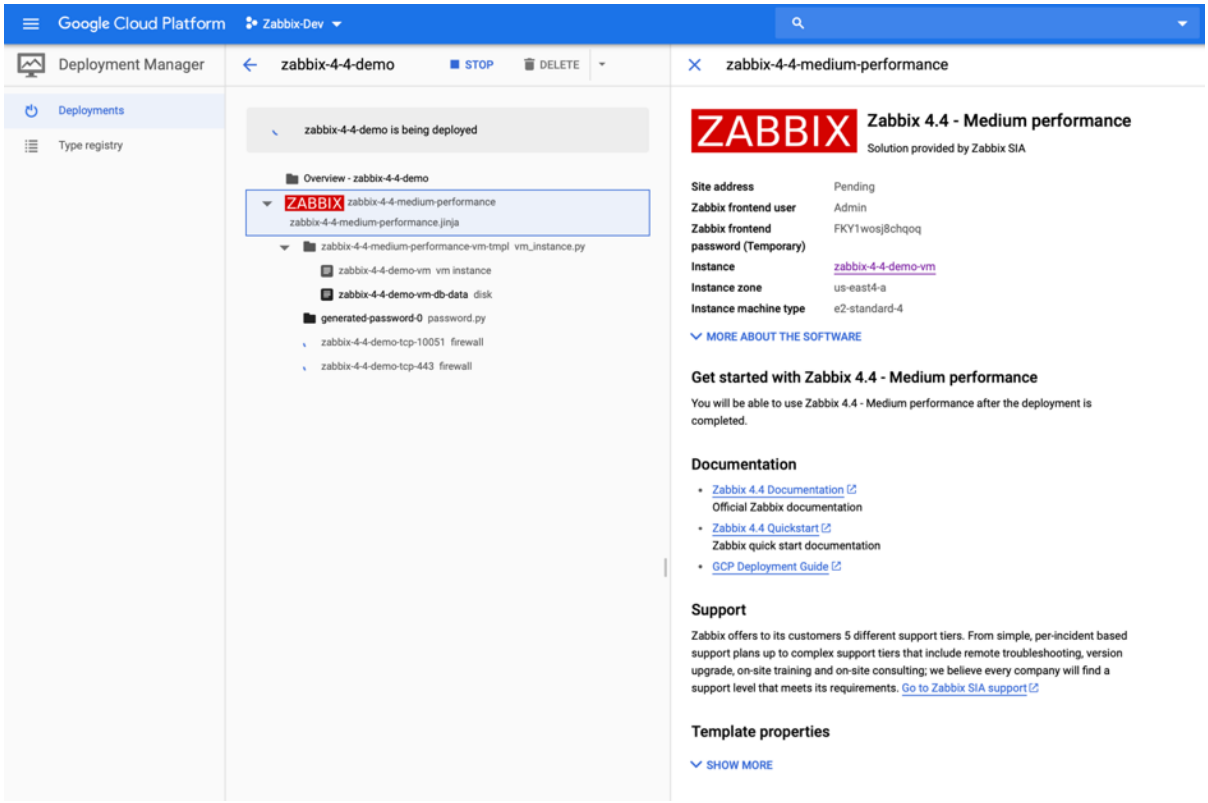
Kliknij *Get started*.



Możesz użyć ustawień domyślnych lub je dostosować. Po zakończeniu kliknij *Deploy*.



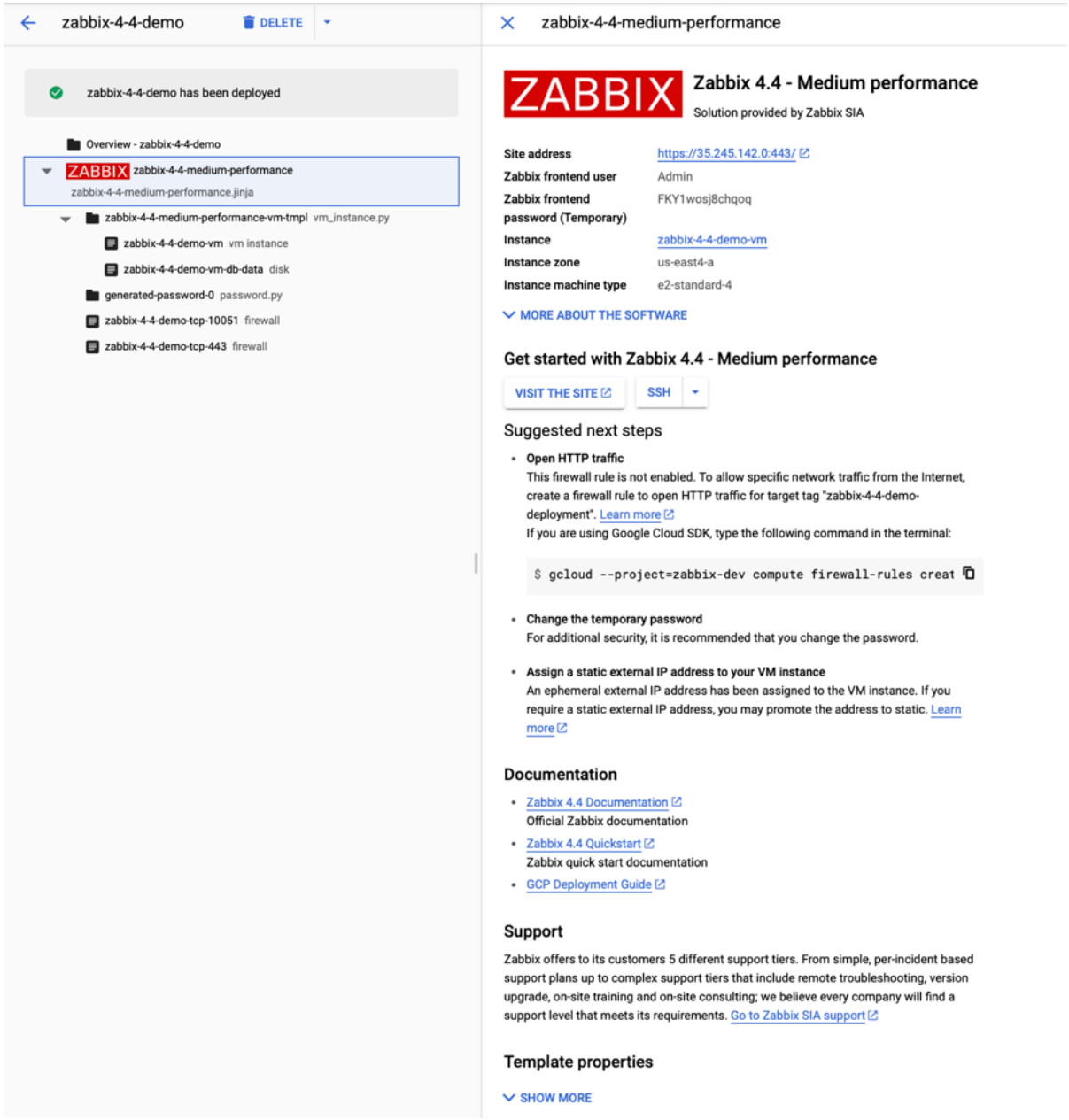
Wdrażanie jest teraz w toku. Poczekaj na jego zakończenie.



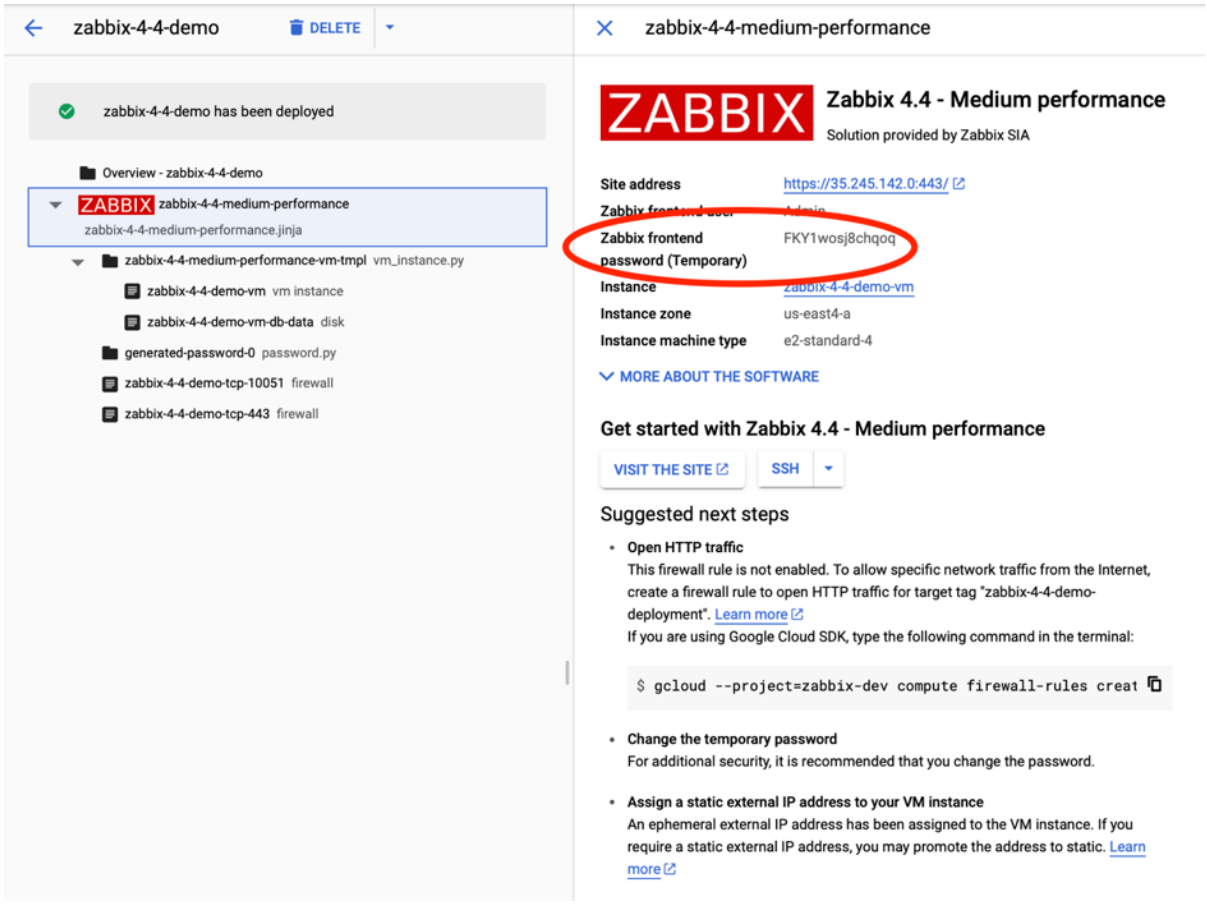
To wszystko! Wdrażanie zostało rozpoczęte.

Uzyskiwanie dostępu do appliance

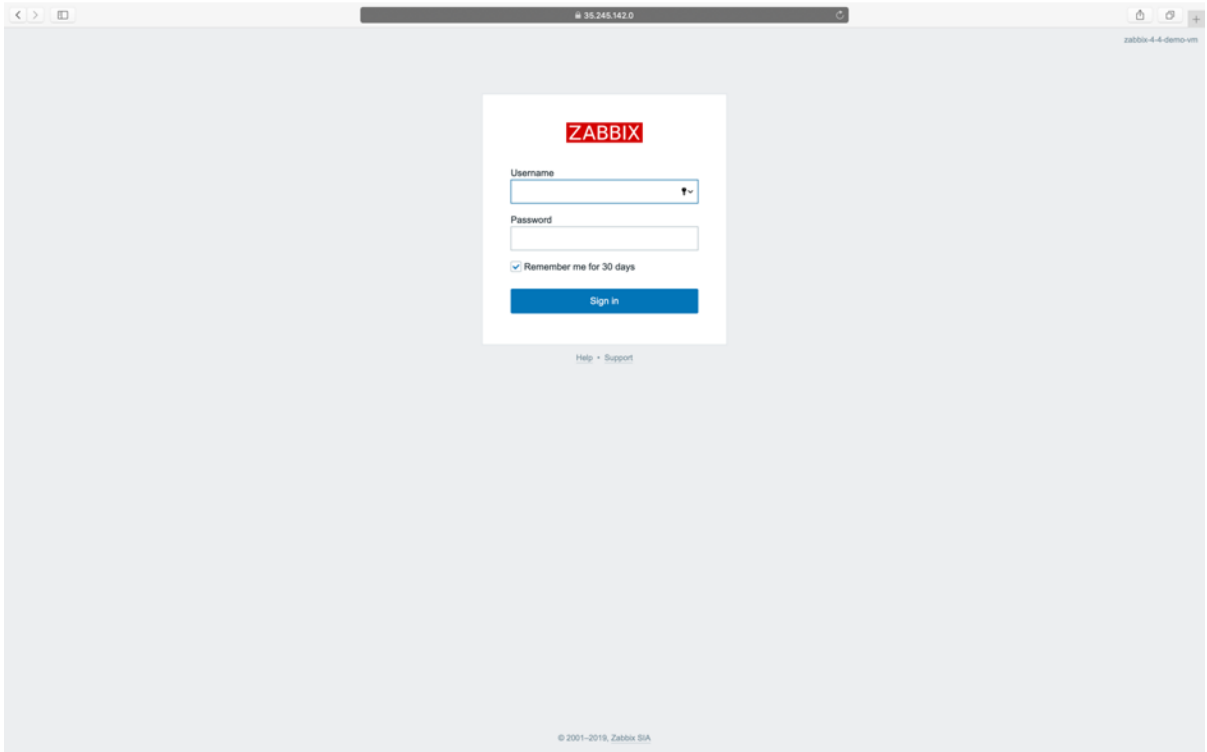
Po zakończeniu wdrażania powinien zostać wyświetlony komunikat informujący, że appliance jest wdrażany.

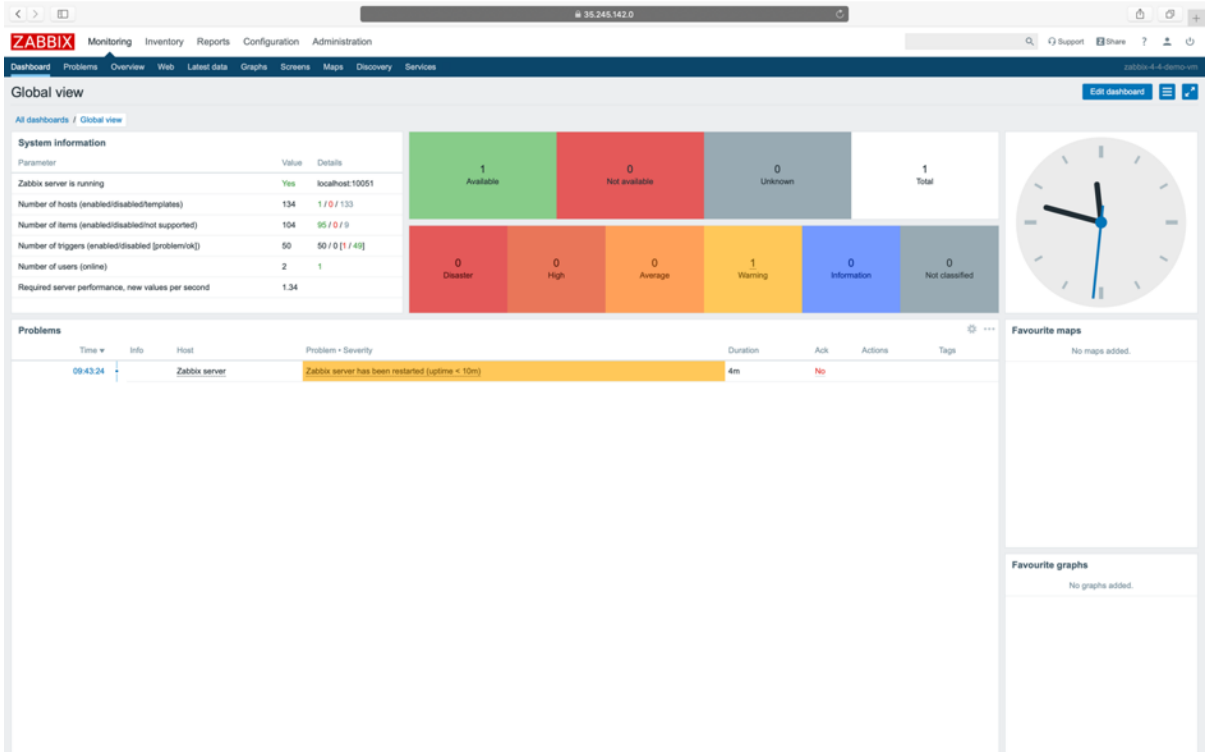


Otwórz przeglądarkę internetową i przejdź do *Site Address*, aby wyświetlić frontend Zabbix. Wcześniej pobierz hasło użytkownika Zabbix *Admin*, aby zalogować się do appliance Zabbix.

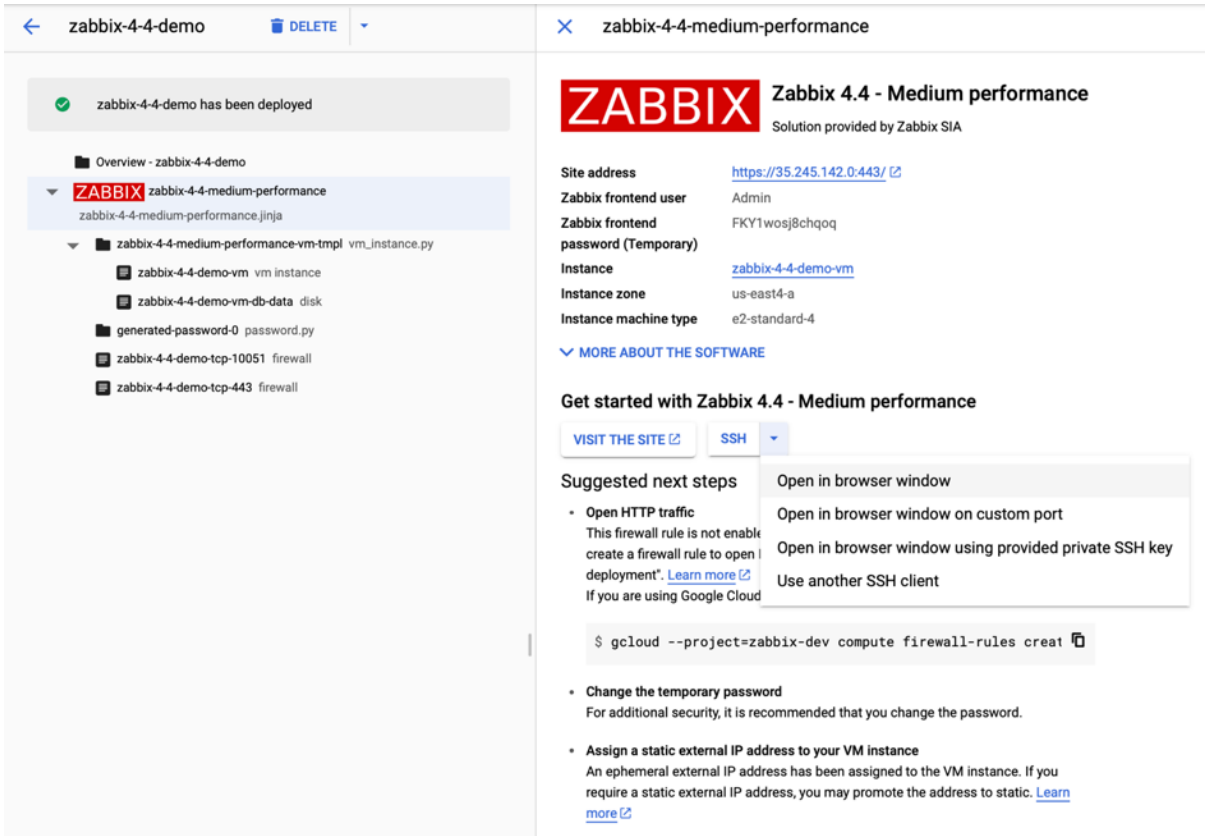


Gotowe! Masz teraz appliance Zabbix.





Możesz również zalogować się do instancji, aby korzystać z bazy danych. W tym celu wróć do konsoli Google i wykonaj czynności wskazane czerwoną strzałką poniżej, aby rozpocząć sesję SSH za pomocą opcji *Open in browser window*.



MySQL jest dostępny bez hasła `root`. Hasło jest przechowywane w pliku konfiguracyjnym `/root/.my.cnf`. Domyślnie appliance Zabbix używa strefy czasowej UTC. Jeśli chcesz ją zmienić, edytuj dyrektywę konfiguracyjną PHP `date.timezone` w pliku `/etc/php.d/99-zabbix.ini`, a następnie uruchom ponownie proces php-fpm: `systemctl restart php-fpm`.

Aktualizacja

Przed aktualizacją zdecydowanie zaleca się zapoznanie ze wszystkimi odpowiednimi uwagami dotyczącymi aktualizacji.

Przydatne będzie również sprawdzenie, które komponenty należy zaktualizować.

Przegląd

Instrukcje aktualizacji krok po kroku są dostępne dla:

- Red Hat Enterprise Linux (przy użyciu pakietów)
- Debian/Ubuntu (przy użyciu pakietów)
- Aktualizacja przy użyciu kontenerów
- Aktualizacja przy użyciu kodu źródłowego

Powiązane instrukcje:

- W przypadku serwerów w klastrze wysokiej dostępności (HA) zobacz Aktualizacja klastra HA
- W przypadku bazy danych TimescaleDB zobacz Aktualizacja schematu TimescaleDB

Komponenty do aktualizacji

Aktualizacja agentów Zabbix jest zalecana, ale nieobowiązkowa.

Aktualizacja proxy Zabbix jest zdecydowanie zalecana.
Serwer Zabbix w pełni obsługuje proxy w tej samej głównej wersji co serwer.
Serwer Zabbix obsługuje również proxy, które **nie są starsze** niż wersja poprzedniego wydania LTS serwera Zabbix, ale z ograniczoną funkcjonalnością (zbieranie danych, wykonywanie poleceń zdalnych, natychmiastowe sprawdzanie wartości pozycji).
Aktualizacja konfiguracji jest również wyłączona, a nieaktualne proxy będą działać tylko ze starą konfiguracją.

Attention:
Proxy starsze niż wersja poprzedniego wydania LTS serwera Zabbix lub nowsze niż główna wersja serwera Zabbix nie są obsługiwane. Serwer Zabbix będzie ignorować dane z nieobsługiwanych proxy, a cała komunikacja z serwerem Zabbix zakończy się niepowodzeniem i wyświetlone zostanie ostrzeżenie. Więcej informacji można znaleźć w sekcji Zgodność wersji.

Aby zminimalizować przestoje i utratę danych podczas aktualizacji, zaleca się zatrzymanie, aktualizację i uruchomienie serwera Zabbix, a następnie kolejno zatrzymywanie, aktualizowanie i uruchamianie proxy Zabbix. Podczas przestoju serwera działające proxy będą kontynuować zbieranie danych. Po uruchomieniu serwera nieaktualne proxy wyślą dane do nowszego serwera (konfiguracja proxy nie zostanie jednak zaktualizowana) i pozostaną częściowo funkcjonalne. Wszelkie powiadomienia dotyczące problemów występujących podczas przestoju serwera Zabbix zostaną wygenerowane dopiero po uruchomieniu zaktualizowanego serwera.

Jeśli proxy Zabbix jest uruchamiany po raz pierwszy, a plik bazy danych SQLite nie istnieje, proxy utworzy go automatycznie.

Uwaga, że jeśli proxy Zabbix korzysta z SQLite3 i podczas uruchamiania wykryje, że wersja istniejącego pliku bazy danych jest starsza niż wymagana, **automatycznie usunie plik bazy danych** i utworzy nowy. W związku z tym dane historyczne przechowywane w pliku bazy danych SQLite zostaną utracone. Jeśli wersja proxy Zabbix jest starsza niż wersja pliku bazy danych, Zabbix zapisze błąd w dzienniku i zakończy działanie.

W zależności od rozmiaru bazy danych aktualizacja bazy danych do wersji 8.0 może potrwać długo.

Uwagi dotyczące aktualizacji

Bezpośrednia aktualizacja do Zabbix 8.0.x jest obsługiwana z wersji **2.0**.x.
W przypadku aktualizacji ze starszych wersji zapoznaj się z dokumentacją Zabbix dotyczącą wersji 2.0 i wcześniejszych.

Note:
Pamiętaj, że po aktualizacji niektóre integracje Zabbix z oprogramowaniem innych firm mogą zostać zakłócone, jeśli oprogramowanie zewnętrzne nie jest zgodne ze zaktualizowaną wersją Zabbix.

Dostępne są następujące informacje dotyczące aktualizacji:

Aktualizacja z	Przeczytaj pełne informacje dotyczące aktualizacji	Najważniejsze zmiany między wersjami
7.4.x	Dla: Zabbix 8.0	Agenty Zabbix w wersji wcześniejszej niż 1.6 nie są już obsługiwane. Podniesiono minimalne wymagane wersje baz danych. Podniesiono minimalną wymaganą wersję PHP z 8.0.0 do 8.2.0. Znak % został dodany do listy `UnsafeUserParameters` dla agenta Zabbix i agenta Zabbix 2.

Aktualizacja z	Przeczytaj pełne informacje dotyczące aktualizacji	Najważniejsze zmiany między wersjami
7.2.x	Dla: Zabbix 7.4 Zabbix 8.0	Usunięto obsługę biblioteki PCRE.
7.0.x	Dla: Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Usunięto obsługę Oracle DB.
6.4.x	Dla: Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Podniesiono minimalną wymaganą wersję PHP z 7.4.0 do 8.0.0. Asynchroniczne pollery dla kontroli agenta, agenta HTTP, SNMP walk[OID] i SNMP get[OID]. Oddzielna tabela bazy danych dla proxy. Zmieniono domyślną lokalizację pliku konfiguracyjnego agenta Windows. Oracle DB jest przestarzała. Usunięto stary typ wartości liczbowej (float).
6.2.x	Dla: Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Podniesiono minimalną wymaganą wersję MySQL z 8.0.0 do 8.0.30. Biblioteka 'libevent_pthreads' jest wymagana dla serwera/proxy Zabbix. Przy pierwszym uruchomieniu po aktualizacji proxy Zabbix z SQLite3 automatycznie usuwa starą wersję bazy danych (wraz z całą historią) i tworzy nową.
6.0.x LTS	Dla: Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Podniesiono minimalną wymaganą wersję PHP z 7.2.5 do 7.4.0. Znacząco zmieniono monitorowanie usług. Podczas aktualizacji należy utworzyć deterministyczne wyzwalacze. Jeśli dla MySQL/MariaDB włączono rejestrowanie binarne, wymaga to uprawnień superużytkownika lub ustawienia zmiennej/parametru konfiguracji *log_bin_trust_function_creators = 1*. Instrukcje ustawiania zmiennej znajdują się w sekcji Skrypty tworzenia bazy danych.
5.4.x	Dla: Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Podniesiono minimalne wymagane wersje baz danych. Serwer/proxy nie uruchomi się, jeśli baza danych jest nieaktualna. Utracono rekordy dziennika audytu z powodu zmiany struktury bazy danych.
5.2.x	Dla: Zabbix 5.4 Zabbix 6.0 Zabbix 6.2 Zabbix 6.4 Zabbix 7.0 Zabbix 7.2 Zabbix 7.4 Zabbix 8.0	Podniesiono minimalne wymagane wersje baz danych. Usunięto pozycje agregujące jako oddzielny typ.
5.0.x LTS	Dla: 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	Podniesiono minimalną wymaganą wersję PHP z 7.2.0 do 7.2.5. Zmieniono algorytm haszowania haseł z MD5 na bcrypt.

Aktualizacja z	Przeczytaj pełne informacje dotyczące aktualizacji	Najważniejsze zmiany między wersjami
4.4.x	Dla: 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	Usunięto obsługę IBM DB2. Podniesiono minimalną wymaganą wersję PHP z 5.4.0 do 7.2.0. Podniesiono minimalne wymagane wersje baz danych. Zmieniono katalog plików PHP Zabbix.
4.2.x	Dla: 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	Usunięto typy multimediów Jabber i Ez Texting.
4.0.x LTS	Dla: 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	Starsze proxy nie mogą już raportować danych do zaktualizowanego serwera. Nowsze agenty nie będą już mogły współpracować ze starszym serwerem Zabbix.
3.4.x	Dla: 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	Biblioteki ’libpthread’ i ’zlib’ są teraz wymagane. Usunięto obsługę protokołu w postaci zwykłego tekstu, a nagłówek jest wymagany. Agenty Zabbix w wersji wcześniejszej niż 1.4 nie są już obsługiwane. Parametr Server w konfiguracji pasywnego proxy jest teraz wymagany.

Aktualizacja z	Przeczytaj pełne informacje dotyczące aktualizacji	Najważniejsze zmiany między wersjami
3.2.x	Dla: 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	Usunięto obsługę SQLite jako bazowej bazy danych dla serwera/frontend Zabbix. Zamiast rozszerzonych wyrażeń regularnych POSIX obsługiwane są wyrażenia regularne zgodne z Perlem (PCRE). Biblioteki 'libpcre' i 'libevent' są wymagane dla serwera Zabbix. Dodano sprawdzanie kodów wyjścia dla parametrów użytkownika, poleceń zdalnych i pozycji system.run[] bez flagi 'nowait', a także skryptów wykonywanych przez serwer Zabbix. Aby obsługiwać nowe funkcje, należy zaktualizować Java gateway Zabbix.
3.0.x LTS	Dla: 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	Aktualizacja bazy danych może trwać długo, zależnie od rozmiaru tabeli historii.
2.4.x	Dla: 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	Podniesiono minimalną wymaganą wersję PHP z 5.3.0 do 5.4.0. Należy określić parametr agenta LogFile.

Aktualizacja z	Przeczytaj pełne informacje dotyczące aktualizacji	Najważniejsze zmiany między wersjami
2.2.x LTS	Dla: 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	Usunięto rozproszone monitorowanie oparte na węzłach.
2.0.x	Dla: 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	Podniesiono minimalną wymaganą wersję PHP z 5.1.6 do 5.3.0. Do prawidłowego działania serwera wymagana jest baza danych MySQL z rozróżnianiem wielkości liter; do prawidłowej współpracy serwera Zabbix z bazą MySQL wymagane są zestaw znaków utf8 i sortowanie utf8_bin. Zobacz skrypty tworzenia bazy danych. Zamiast rozszerzenia PHP ’mysql’ wymagane jest rozszerzenie ’mysqli’.

Upgrade from packages

Przegląd

Ta sekcja zawiera kroki wymagane do pomyślnego uaktualnienia przy użyciu oficjalnych pakietów RPM i DEB dostarczanych przez Zabbix dla:

- Red Hat Enterprise Linux
- Debian/Ubuntu

Pakiety Zabbix z repozytoriów systemu operacyjnego

Niektóre dystrybucje systemów operacyjnych (w szczególności dystrybucje oparte na Debianie) udostępniają własne pakiety Zabbix. Pakiety te **nie są wspierane przez Zabbix** i mogą być nieaktualne lub nie zawierać najnowszych funkcji oraz poprawek błędów. Zaleca się korzystanie wyłącznie z oficjalnych pakietów z Oficjalnego Repozytorium Zabbix.

Jeśli przeprowadzasz aktualizację z pakietów dostarczanych przez dystrybucje systemu operacyjnego (lub były one zainstalowane w pewnym momencie), wykonaj poniższą procedurę, aby przejść na oficjalne pakiety Zabbix:

1. Odinstaluj stare pakiety.
2. Sprawdź, czy po starej instalacji nie pozostały żadne pliki, i usuń je.
3. Zainstaluj oficjalne pakiety Zabbix zgodnie z instrukcjami instalacji udostępnionymi przez Zabbix.

Nie wykonuj bezpośredniej aktualizacji, ponieważ może to spowodować uszkodzenie instalacji.

Debian/Ubuntu

Przegląd

Ta sekcja zawiera instrukcje dotyczące aktualizacji z Zabbix **7.4.x** do najnowszej wersji Zabbix **8.0.x** przy użyciu oficjalnych pakietów Zabbix dla Debiana/Ubuntu.

Warning:
Przed aktualizacją zapoznaj się z odpowiednimi uwagami dotyczącymi aktualizacji i upewnij się, że Twój system spełnia wymagania dla Zabbix 8.0.

Note:
Rozważ uruchomienie dwóch równoległych sesji SSH podczas aktualizacji: jednej do wykonywania kroków aktualizacji, a drugiej do monitorowania logów serwera/proxy. Na przykład w drugiej sesji uruchom `tail -f zabbix_server.log` lub `tail -f zabbix_proxy.log`, aby wyświetlać najnowsze wpisy logów i możliwe błędy w czasie rzeczywistym. Może to mieć kluczowe znaczenie w środowiskach produkcyjnych.

Instrukcje dotyczące aktualizacji między wersjami podrzędnymi Zabbix 8.0.x (na przykład z 8.0.1 do 8.0.3) znajdują się w sekcji Aktualizacja między wersjami podrzędnymi.

Procedura aktualizacji

1 Zatrzymaj procesy Zabbix

Zatrzymaj serwer Zabbix, aby upewnić się, że do bazy danych nie są wstawiane żadne nowe dane:

```
systemctl stop zabbix-server
```

Jeśli aktualizujesz Zabbix proxy, agent lub agent 2, zatrzymaj również te komponenty:

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2 Utwórz kopię zapasową bazy danych Zabbix

Utwórz kopię zapasową istniejącej bazy danych Zabbix, aby zabezpieczyć się przed niepowodzeniem aktualizacji (na przykład z powodu braku miejsca na dysku, utraty zasilania lub nieoczekiwanych problemów).

3 Utwórz kopię zapasową plików konfiguracyjnych Zabbix, plików PHP i binariów Zabbix

Utwórz kopię zapasową istniejących plików konfiguracyjnych Zabbix, plików PHP i binariów Zabbix.

W przypadku plików konfiguracyjnych uruchom:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
##### Dla Apache:
cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-backup/
##### Dla Nginx:
cp /etc/nginx/conf.d/zabbix.conf /opt/zabbix-backup/
cp /etc/zabbix/nginx.conf /opt/zabbix-backup/
```

W przypadku plików PHP i binariów Zabbix uruchom:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Zaktualizuj pakiet konfiguracji repozytorium

Przed kontynuowaniem aktualizacji odinstaluj bieżący pakiet repozytorium Zabbix:

```
rm -Rf /etc/apt/sources.list.d/zabbix.list
```

Może być również konieczne ręczne usunięcie starych pakietów Zabbix z katalogu roboczego (np. `rm zabbix-release_latest+debian12_` przed pobraniem nowego, aby zapobiec ponownemu użyciu przez menedżera pakietów nieaktualnej wersji podczas procesu aktualizacji.

Następnie zainstaluj najnowszy pakiet konfiguracji repozytorium, aby zapewnić zgodność z najnowszymi pakietami oraz uwzględnić wszelkie ostatnie poprawki bezpieczeństwa lub błędy.

W przypadku **Debian 13** uruchom:

```
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
```

W przypadku **Debian 12** uruchom:

```
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:**
> W przypadku starszych wersji Debiana zastąp powyższy link właściwym linkiem z repozytorium Zabbix. Należy jednak pamiętać, że pakiety dla tych wersji mogą nie zawierać wszystkich komponentów Zabbix, a aby zaktualizować te komponenty z pakietów, rozważ aktualizację systemu operacyjnego. Listę dołączonych komponentów znajdziesz w sekcji Zabbix packages.

W przypadku **Ubuntu 26.04** uruchom:

```
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
```

W przypadku **Ubuntu 24.04** uruchom:

```
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
```

W przypadku **Ubuntu 22.04** uruchom:

```
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:**
> W przypadku starszych wersji Ubuntu zastąp powyższy link właściwym linkiem z repozytorium Zabbix. Należy jednak pamiętać, że pakiety dla tych wersji mogą nie zawierać wszystkich komponentów Zabbix, a aby zaktualizować te komponenty z pakietów, rozważ aktualizację systemu operacyjnego. Listę dołączonych komponentów znajdziesz w sekcji Zabbix packages.

Może pojawić się monit dotyczący konfiguracji repozytorium 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] ?
```

Wpisz `Y` (lub `I`), aby zainstalować wersję pliku konfiguracyjnego repozytorium Zabbix dostarczoną przez opiekuna pakietu.

Następnie zaktualizuj informacje o repozytorium:

```
apt update
```

5 Aktualizacja komponentów Zabbix

Aby zaktualizować komponenty Zabbix, uruchom:

```
apt install --only-upgrade zabbix-server-mysql zabbix-frontend-php zabbix-agent
```

- Jeśli używasz PostgreSQL, zamień `mysql` na `pgsql` w poleceniu.
- Jeśli aktualizujesz proxy, zamień `server` na `proxy` w poleceniu.
- Jeśli aktualizujesz agenta Zabbix 2, zamień `zabbix-agent` na `zabbix-agent2 zabbix-agent2-plugin-*` w poleceniu.

Attention:
Aktualizacja agenta Zabbix 2 za pomocą polecenia `apt install zabbix-agent2` może spowodować błąd. Więcej informacji znajdziesz w sekcji *Znane problemy*.

Może pojawić się monit dotyczący konfiguracji serwera Zabbix (lub 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] ?
```

Wprowadź opcję, która najlepiej odpowiada Twojej sytuacji. Na przykład wprowadź `D`, aby porównać bieżącą i nową konfigurację, a następnie zdecyduj, czy chcesz zainstalować wersję opiekuna pakietu (`Y` lub `I`).

Następnie zaktualizuj frontend Zabbix i uruchom ponownie serwer WWW.

W przypadku Apache uruchom:

```
apt install zabbix-apache-conf
systemctl restart apache2
```

W przypadku Nginx uruchom:

```
apt install zabbix-nginx-conf
systemctl restart nginx
```

W przypadku Nginx może pojawić się monit dotyczący pliku konfiguracyjnego `nginx.conf`. Wprowadź opcję, która najlepiej odpowiada Twojej sytuacji:

```
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] ?
```

Jeśli po aktualizacji musisz zainstalować dodatkowy pakiet Zabbix z tego samego repozytorium, `apt` może podczas rozwiązywania zależności wybrać aktualizację już zainstalowanych pakietów Zabbix. Aby uniknąć aktualizacji zainstalowanego pakietu serwera Zabbix lub proxy, tymczasowo wstrzymaj ten pakiet przed instalacją, a następnie usuń wstrzymanie:

```
apt-mark hold zabbix-server-mysql
apt install zabbix-sender
apt-mark unhold zabbix-server-mysql
```

6 Przejrzyj parametry konfiguracji komponentu

Przejrzyj odpowiednie uwagi dotyczące aktualizacji, aby sprawdzić, czy konieczne są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdują się na stronie Co nowego.

Jeśli jako serwer WWW używasz Nginx, a plik konfiguracyjny `/etc/zabbix/nginx.conf` został ponownie zainstalowany podczas aktualizacji, przejrzyj ten plik i zaktualizuj dyrektywy `listen` oraz `server_name`, aby odpowiadały konfiguracji frontend Zabbixa.

7 Uruchom procesy Zabbix

Uruchom zaktualizowane komponenty Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
```

```
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8 Wyczyść pliki cookie i pamięć podręczną przeglądarki internetowej

Po aktualizacji może być konieczne wyczyszczenie plików cookie i pamięci podręcznej przeglądarki internetowej, aby interfejs webowy Zabbix działał prawidłowo.

Aktualizacja między wersjami minor

Możliwa jest aktualizacja między wersjami minor Zabbix 8.0.x (na przykład z 8.0.1 do 8.0.3).

Najpierw zaktualizuj informacje o repozytorium:

```
apt update
```

Następnie, aby zaktualizować wszystkie komponenty Zabbix, uruchom:

```
apt install --only-upgrade 'zabbix*'
```

- Aby zaktualizować tylko serwer Zabbix, zastąp w poleceniu `'zabbix*'` przez `'zabbix-server*'`.
- Aby zaktualizować tylko proxy Zabbix, zastąp w poleceniu `'zabbix*'` przez `'zabbix-proxy*'`.
- Aby zaktualizować tylko agent Zabbix, zastąp w poleceniu `'zabbix*'` przez `'zabbix-agent*'`.
- Aby zaktualizować tylko agent 2 Zabbix, zastąp w poleceniu `'zabbix*'` przez `'zabbix-agent2*'`.

Red Hat Enterprise Linux

Przegląd

Ta sekcja zawiera instrukcje dotyczące aktualizacji z Zabbix **7.4.x** do najnowszej wersji Zabbix **8.0.x** przy użyciu oficjalnych pakietów Zabbix dla Red Hat Enterprise Linux lub jego pochodnych — AlmaLinux, CentOS Stream, Oracle Linux i Rocky Linux.

Warning:
Przed aktualizacją zapoznaj się z odpowiednimi uwagami dotyczącymi aktualizacji i upewnij się, że Twój system spełnia wymagania dla Zabbix 8.0.

Note:
Rozważ uruchomienie dwóch równoległych sesji SSH podczas aktualizacji: jednej do wykonywania kroków aktualizacji, a drugiej do monitorowania logów serwera/proxy. Na przykład uruchom `tail -f zabbix_server.log` lub `tail -f zabbix_proxy.log` w drugiej sesji, aby wyświetlać najnowsze wpisy logów i ewentualne błędy w czasie rzeczywistym. Może to mieć kluczowe znaczenie w przypadku instancji produkcyjnych.

Instrukcje dotyczące aktualizacji między wersjami podrzędnymi Zabbix 8.0.x (na przykład z 8.0.1 do 8.0.3) znajdują się w sekcji Aktualizacja między wersjami podrzędnymi.

Procedura aktualizacji

1 Zatrzymaj procesy Zabbix

Zatrzymaj serwer Zabbix, aby upewnić się, że żadne nowe dane nie są wstawiane do bazy danych:

```
systemctl stop zabbix-server
```

Jeśli aktualizujesz Zabbix proxy, agent lub agent 2, zatrzymaj również te komponenty:

```
systemctl stop zabbix-proxy
systemctl stop zabbix-agent
systemctl stop zabbix-agent2
```

2 Utwórz kopię zapasową bazy danych Zabbix

Utwórz kopię zapasową istniejącej bazy danych Zabbix, aby zabezpieczyć się przed niepowodzeniem aktualizacji (na przykład z powodu problemów z miejscem na dysku, utraty zasilania lub nieoczekiwanych problemów).

3 Utwórz kopię zapasową plików konfiguracyjnych Zabbix, plików PHP i plików binarnych Zabbix

Utwórz kopię zapasową istniejących plików konfiguracyjnych Zabbix, plików PHP i plików binarnych Zabbix.

Dla plików konfiguracyjnych uruchom:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
cp /etc/httpd/conf.d/zabbix.conf  /opt/zabbix-backup/
```

Dla plików PHP i plików binarnych Zabbix uruchom:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Aktualizacja pakietu konfiguracji repozytorium

Przed rozpoczęciem aktualizacji zaktualizuj bieżący pakiet repozytorium do najnowszej wersji, aby zapewnić zgodność z najnowszymi pakietami oraz uwzględnić najnowsze poprawki zabezpieczeń i błędów.

W systemie **RHEL 10** uruchom:

```
rpm -Uvh https://repo.zabbix.com/zabbix/8.0/release/rhel/10/noarch/zabbix-release-latest.el10.noarch.rpm
```

Jeśli aktualizujesz proxy Zabbix w systemie RHEL 10 i chcesz używać APM, zainstaluj również repozytorium EPEL i włącz repozytorium CRB. Oba repozytoria udostępniają zależności dla pakietu APM proxy Zabbix:

```
dnf -y install epel-release || dnf -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-$(
crb enable
```

W systemie **RHEL 9** uruchom:

```
rpm -Uvh https://repo.zabbix.com/zabbix/8.0/release/rhel/9/noarch/zabbix-release-latest.el9.noarch.rpm
```

> **Note:**
> W przypadku starszych wersji RHEL lub ich pochodnych zastąp powyższy link właściwym linkiem z repozytorium Zabbix. Pamiętaj jednak, że pakiety dla tych wersji mogą nie zawierać wszystkich komponentów Zabbix. Aby zaktualizować te komponenty z pakietów, rozważ aktualizację systemu operacyjnego. Listę zawartych komponentów znajdziesz na stronie Pakiety Zabbix.

Następnie wyczyść pamięć podręczną menedżera pakietów `dnf` (w tym nagłówki, metadane i pliki pakietów pobrane podczas wcześniejszych instalacji lub aktualizacji):

```
dnf clean all
```

Podczas następnej operacji `dnf` pobierze świeże metadane z repozytoriów, ponieważ stare metadane zostały usunięte.

Zobacz także: Znane problemy, aby dowiedzieć się, jak zaktualizować pakiet konfiguracji repozytorium w systemie RHEL.

5 Aktualizacja komponentów Zabbix

Aby zaktualizować komponenty Zabbix, uruchom:

```
dnf install zabbix-server-mysql zabbix-web-mysql zabbix-selinux-policy zabbix-agent
```

- Jeśli używasz PostgreSQL, zamień `mysql` na `pgsql` w poleceniu.
- Jeśli aktualizujesz proxy, zamień `server` na `proxy` w poleceniu.
- Jeśli masz RHEL 10 i chcesz używać APM, dodaj `-apm` do nazwy pakietu proxy (np. `zabbix-proxy-mysql-apm`).
- Jeśli aktualizujesz agent 2, zamień `zabbix-agent` na `zabbix-agent2` `zabbix-agent2-plugin-*` w poleceniu.
- Jeśli masz zainstalowane inne pakiety Zabbix, rozważ ich aktualizację, dodając je do polecenia.

> **Attention:**
> Aktualizacja agent 2 za pomocą polecenia `dnf install zabbix-agent2` może spowodować błąd. Więcej informacji znajdziesz w sekcji *Znane problemy*.

Następnie, aby zaktualizować frontend Zabbix z Apache i ponownie uruchomić Apache, uruchom:

```
dnf install zabbix-apache-conf
systemctl restart httpd
```

6 Przejrzyj parametry konfiguracji komponentu

Przejrzyj odpowiednie uwagi dotyczące aktualizacji, aby sprawdzić, czy konieczne są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

7 Uruchom procesy Zabbix

Uruchom zaktualizowane komponenty Zabbix:

```
systemctl start zabbix-server
systemctl start zabbix-proxy
systemctl start zabbix-agent
systemctl start zabbix-agent2
```

8 Wyczyść pliki cookie i pamięć podręczną przeglądarki internetowej

Po aktualizacji może być konieczne wyczyszczenie plików cookie i pamięci podręcznej przeglądarki internetowej, aby interfejs webowy Zabbix działał prawidłowo.

Aktualizacja między wersjami pobocznymi

Możliwa jest aktualizacja między wersjami pobocznymi Zabbix 8.0.x (na przykład z 8.0.1 do 8.0.3).

Aby zaktualizować wszystkie komponenty Zabbix, uruchom:

```
dnf upgrade 'zabbix-*'
```

- Aby zaktualizować tylko serwer Zabbix, zastąp w poleceniu `'zabbix-*'` przez `'zabbix-server-*'`.
- Aby zaktualizować tylko proxy Zabbix, zastąp w poleceniu `'zabbix-*'` przez `'zabbix-proxy-*'`.
- Aby zaktualizować tylko agent Zabbix, zastąp w poleceniu `'zabbix-*'` przez `'zabbix-agent-*'`.
- Aby zaktualizować tylko Zabbix agent 2, zastąp w poleceniu `'zabbix-*'` przez `'zabbix-agent2-*'`.

Problemy podczas aktualizacji

Związane z sekwencjami ucieczki Przekroczony limit znaków dla parametrów funkcji po aktualizacji

W Zabbix 7.0 dodano prawidłowe stosowanie znaków ucieczki dla ukośników odwrotnych w parametrach tekstowych funkcji historii.

Ponieważ podczas aktualizacji z wersji Zabbix starszych niż 7.0 dodawane są dodatkowe ukośniki odwrotne, prowadzi to do wydłużenia parametrów, co może skutkować uszkodzeniem funkcji wyzwalaczy, jeśli długość parametru przekroczy maksymalny rozmiar danych wynoszący 255 znaków.

Aby zilustrować, jakie problemy mogą wystąpić podczas aktualizacji bazy danych, jaki mają wpływ oraz jak je rozpoznać i naprawić, rozważmy przykładową konfigurację z 3 pozycjami i 2 wyzwalaczami.

Konfiguracja

1. Wersja Zabbix 6.0.

2. Plik `/tmp/ONE`, który zawiera 114 ukośników odwrotnych, z literą „a" na końcu:

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

3. Pozycja `vfs.file.contents[/tmp/ONE]` do odczytu pliku /tmp/ONE.

4. Dwie pozycje kalkulowane:

Calc_228 - jej parametr zawiera 228 ukośników odwrotnych, które są poprzedzone znakiem ucieczki, więc **pasują** do 114 ukośników odwrotnych z pliku /tmp/ONE:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

Calc_232 - jej parametr zawiera 232 ukośniki odwrotne, które są poprzedzone znakiem ucieczki, więc **nie pasują** do 114 ukośników odwrotnych z pliku /tmp/ONE, ponieważ wymagają co najmniej 116 rzeczywistych ukośników odwrotnych:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

5. Dwa wyzwalacze:

Trig_228 - podobnie jak w przypadku Calc_228, jego parametr zawiera 228 ukośników odwrotnych, aby **dopasować** 114 rzeczywistych ukośników odwrotnych z pliku /tmp/ONE:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

Trig_232 - podobnie jak w przypadku Calc_232, jego parametr zawiera 232 ukośniki odwrotne, aby dopasować 116 rzeczywistych ukośników odwrotnych, więc **nie pasuje** do 114 ukośników odwrotnych z pliku /tmp/ONE:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1h:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

6. Calc_228 zwraca 1, a Trig_228 **uruchamia się**.
7. Calc_232 zwraca 0, a Trig_232 **nie uruchamia się**.

Aktualizacja

1. Uruchom aktualizację do wersji 8.0.0 i sprawdź logi pod kątem ostrzeżenia:

```
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:
'$,1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\a"'
Resulting escaped value would be:
'$,1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
DB upgrade and Zabbix server can continue to run, but to make sure this function works correctly
MANUAL INTERVENTION IS REQUIRED !
Need to manually reduce the size of this parameter with macro as a workaround.

2485502:20250228:115442.237 DBpatch_6050165(): cannot save in DB function parameter: resulting size 485 is
functionid:33795 function:'find'
used on host: 'Zabbix server'
  in trigger: 'TRIG_232'.
Current parameter value:
'$,1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\a"'
Resulting escaped value would be:
'$,1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
DB upgrade and Zabbix server can continue to run, but to make sure this function works correctly
MANUAL INTERVENTION IS REQUIRED !
Need to manually reduce the size of this parameter with macro as a workaround.
```

2. Aktualizacja bazy danych **kończy się**, a serwer Zabbix nadal działa.
3. Calc_228 nadal zwraca 1, podczas gdy Calc_232 zwraca 0.
4. Wyzwalacze zostały zaktualizowane w następujący sposób:

Trig_228:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

Trig_232:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp","\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

Oba wyzwalacze — Trig_228 i Trig_232 — **teraz się uruchamiają**! Jest to nieoczekiwane, więc wymagana jest ręczna interwencja.

Ręczna interwencja

Wyrażenia wyzwalaczy muszą zostać zaktualizowane tak, aby ich ukośniki odwrotne były poprzedzone znakiem ucieczki, jednak nie można tego zrobić, ponieważ wynikowe parametry byłyby zbyt długie.

Dlatego należy dodać makra:

{$228_BACKSLASHES_A}:

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

{$232_BACKSLASHES_A}:

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

oraz **zaktualizować** wyrażenia wyzwalaczy, aby ich używały:

Trig_228:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp",{$228_BACKSLASHES_A})
```

Trig_232:

```
find(/Zabbix server/vfs.file.contents[/tmp/ONE],1m:now,"regexp",{$232_BACKSLASHES_A})
```

Teraz oba wyzwalacze działają tak samo jak wcześniej — tylko Trig_228 jest **uruchamiany**.

Nie ma problemów z pozycjami obliczanymi, ale dla zachowania spójności również można je zaktualizować:

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})
```

Upgrade from containers

Przegląd

Ta sekcja opisuje kroki wymagane do pomyślnego uaktualnienia kontenerów Zabbix **8.0**.x.

Dostępne są oddzielne zestawy instrukcji dla:

- Uaktualnienie obrazu komponentu Zabbix
- Uaktualnienie pliku Docker Compose

Warning:
Przed uaktualnieniem należy zapoznać się z odpowiednimi **uwagami do uaktualnienia**!

Attention:
Przed rozpoczęciem uaktualnienia należy sprawdzić, czy użytkownicy mają niezbędne uprawnienia do bazy danych na potrzeby uaktualnienia.

 W przypadku uaktualnienia z Zabbix 6.0 lub starszego podczas uaktualnienia konieczne będzie utworzenie deterministycznych wyzwalaczy. Jeśli dla MySQL/MariaDB włączone jest logowanie binarne, wymaga to uprawnień superużytkownika lub ustawienia zmiennej/parametru konfiguracyjnego `log_bin_trust_function_creators = 1`. Instrukcje dotyczące ustawienia tej zmiennej znajdują się w sekcji Skrypty tworzenia bazy danych.

 Należy pamiętać, że w przypadku wykonania z konsoli zmienna zostanie ustawiona tylko tymczasowo i zostanie usunięta po ponownym uruchomieniu Dockera. W takim przypadku należy pozostawić usługę SQL uruchomioną, zatrzymać tylko usługę `zabbix-server`, uruchamiając `docker compose down zabbix-server`, a następnie `docker compose up -d zabbix-server`.

 Alternatywnie można ustawić tę zmienną w pliku konfiguracyjnym.

W zależności od rozmiaru bazy danych uaktualnienie bazy danych do wersji 8.0 może zająć dużo czasu.

Aktualizacja obrazu komponentu Zabbix

Te instrukcje służą do aktualizacji obrazów Docker używanych jako baza dla kontenerów komponentów Zabbix.

Poniższe kroki wykorzystują proxy Zabbix jako przykład. Zastąp `zabbix-proxy-sqlite3` nazwą kontenera swojego komponentu.

1. Sprawdź wszystkie kontenery, aby zidentyfikować ich nazwy i bieżące wersje obrazów:

```
docker ps -a

#### CONTAINER ID   IMAGE                                          (...) NAMES
#### 5a0c2e3d02b4   zabbix/zabbix-proxy-sqlite3:alpine-7.4-latest   (...) zabbix-proxy-sqlite3
```

2. Pobierz nową wersję obrazu:

```
docker pull zabbix/zabbix-proxy-sqlite3:alpine-8.0-latest
```

3. Zatrzymaj kontener komponentu Zabbix:

```
docker stop zabbix-proxy-sqlite3
```

4. Usuń kontener komponentu Zabbix:

Attention:
Upewnij się, że zanotowano wszystkie flagi konfiguracji i punkty montowania woluminów użyte w oryginalnym poleceniu `docker run`, zanim przejdziesz dalej, ponieważ będą one potrzebne do odtworzenia kontenera.

```
docker rm zabbix-proxy-sqlite3
```

5. Uruchom nowy kontener z użyciem zaktualizowanego obrazu:

```
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:
Więcej informacji o uruchamianiu kontenerów Zabbix znajdziesz w sekcji Docker (ręczne wdrożenie).

6. Sprawdź, czy aktualizacja zakończyła się powodzeniem, analizując logi kontenera:

```
docker logs -f zabbix-proxy-sqlite3
```

Aktualizacja pliku Docker Compose

Te instrukcje służą do aktualizacji plików Docker Compose używanych do wdrażania komponentów Zabbix. Postępuj zgodnie z tymi instrukcjami, jeśli Zabbix został zainstalowany przy użyciu Docker Compose.

1. Sprawdź wszystkie kontenery, aby zidentyfikować ich nazwy i bieżące wersje obrazów:

```
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-ngina
#### 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. Przejdź do repozytorium Zabbix Docker, które zostało wcześniej sklonowane, pobierz najnowsze aktualizacje i przełącz się na gałąź 8.0:

Attention:
Jeśli zmodyfikowano plik `.env`, `compose.yaml` lub inne pliki w lokalnym repozytorium, polecenie `git pull` może je nadpisać. Przed wykonaniem tego kroku wykonaj kopię zapasową wszystkich lokalnych zmian.

```
cd zabbix-docker
git pull
git checkout 8.0
```

3. Wdróż komponenty Zabbix, używając zaktualizowanego pliku Docker Compose:

```
#### Z MySQL jako bazą danych:
docker compose -f ./compose.yaml up -d

#### Z PostgreSQL jako bazą danych:
docker compose -f ./compose_pgsql.yaml up -d
```

Note:
Po ponownym utworzeniu i uruchomieniu wszystkich kontenerów wznowienie zbierania danych może potrwać 1–3 minuty.
Więcej informacji o uruchamianiu kontenerów Zabbix można znaleźć w sekcji Docker Compose.

4. Sprawdź, czy aktualizacja zakończyła się powodzeniem, analizując logi kontenera (zastąp `zabbix-docker-zabbix-server-1` nazwą kontenera swojego komponentu):

```
docker logs -f zabbix-docker-zabbix-server-1
```

Upgrade from sources

Przegląd

W tej sekcji przedstawiono kroki wymagane do pomyślnej aktualizacji z Zabbix **7.4**.x do Zabbix **8.0**.x przy użyciu oficjalnych źródeł Zabbix.

> **Warning:**
> Przed aktualizacją należy zapoznać się z odpowiednimi **uwagami dotyczącymi aktualizacji**!

Warto również zapoznać się z wymaganiami dla wersji 8.0.

> **Note:**
> Podczas aktualizacji przydatne może być uruchomienie dwóch równoległych sesji SSH: w jednej należy wykonywać kroki aktualizacji, a w drugiej monitorować logi serwera/proxy. Na przykład w drugiej sesji SSH można uruchomić polecenie `tail -f zabbix_server.log` lub `tail -f zabbix_proxy.log`, aby wyświetlać najnowsze wpisy w pliku logu i możliwe błędy w czasie rzeczywistym. Może to mieć kluczowe znaczenie w przypadku instancji produkcyjnych.

Proces aktualizacji serwera

1 Zatrzymaj serwer

Zatrzymaj serwer Zabbix, aby upewnić się, że do bazy danych nie są wstawiane nowe dane.

2 Utwórz kopię zapasową istniejącej bazy danych Zabbix

To bardzo ważny krok.
Upewnij się, że masz kopię zapasową bazy danych.
Pomoże ona w przypadku niepowodzenia procedury aktualizacji (brak miejsca na dysku, wyłączenie zasilania lub inny nieoczekiwany problem).

3 Tworzenie kopii zapasowej plików konfiguracyjnych, plików PHP i plików binarnych Zabbix

Utwórz kopię zapasową plików binarnych Zabbix, plików konfiguracyjnych oraz katalogu plików PHP.

4 Instalowanie nowych plików binarnych serwera

Skorzystaj z tych instrukcji, aby skompilować serwer Zabbix ze źródeł.

5 Przejrzyj parametry konfiguracji serwera

Upewnij się, że zapoznałeś się z Uwagi dotyczące aktualizacji, aby sprawdzić, czy wymagane są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

6 Uruchom nowe pliki binarne Zabbix

Uruchom nowe pliki binarne.
Sprawdź pliki dziennika, aby upewnić się, że pliki binarne zostały uruchomione pomyślnie.

serwer Zabbix automatycznie zaktualizuje bazę danych.
Podczas uruchamiania serwer Zabbix zgłasza bieżące (obowiązkowe i opcjonalne) oraz wymagane wersje bazy danych.
Jeśli bieżąca wersja obowiązkowa jest starsza niż wymagana wersja, serwer Zabbix automatycznie wykonuje wymagane poprawki aktualizujące bazę danych.
Informacje o rozpoczęciu aktualizacji bazy danych i jej postępie (w procentach) są zapisywane w pliku dziennika serwera Zabbix.
Po zakończeniu aktualizacji w pliku dziennika zapisywany jest komunikat „database upgrade fully completed”.
Jeśli którakolwiek z poprawek aktualizujących zakończy się niepowodzeniem, serwer Zabbix nie uruchomi się.
serwer Zabbix nie uruchomi się również, jeśli bieżąca obowiązkowa wersja bazy danych jest nowsza niż wymagana.
serwer Zabbix uruchomi się tylko wtedy, gdy bieżąca obowiązkowa wersja bazy danych odpowiada wymaganej wersji obowiązkowej.

```
8673:20161117:104750.259 current database version (mandatory/optional): 03040000/03040000
8673:20161117:104750.259 required mandatory version: 03040000
```

Przed uruchomieniem serwera:

- Upewnij się, że użytkownik bazy danych ma wystarczające uprawnienia (tworzenie tabel, usuwanie tabel, tworzenie indeksów, usuwanie indeksów)
- Upewnij się, że masz wystarczającą ilość wolnego miejsca na dysku.

7 Instalacja nowego frontend Zabbix

Minimalna wymagana wersja PHP to 8.2.0. W razie potrzeby zaktualizuj PHP i postępuj zgodnie z instrukcjami instalacji.

8 Wyczyść pliki cookie i pamięć podręczną przeglądarki internetowej

Po aktualizacji może być konieczne wyczyszczenie plików cookie i pamięci podręcznej przeglądarki internetowej, aby frontend Zabbix działał prawidłowo.

Proces aktualizacji proxy

1 Zatrzymaj proxy

Zatrzymaj Zabbix proxy.

2 Utwórz kopię zapasową plików konfiguracyjnych i plików binarnych Zabbix proxy

Utwórz kopię zapasową pliku binarnego Zabbix proxy oraz pliku konfiguracyjnego.

3 Instalowanie nowych plików binarnych proxy

Skorzystaj z tych instrukcji, aby skompilować proxy Zabbix ze źródeł.

4 Przejrzyj parametry konfiguracji proxy

Upewnij się, że przejrzysz Uwagi dotyczące aktualizacji, aby sprawdzić, czy wymagane są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

5 Uruchom nowy proxy Zabbix

Uruchom nowy proxy Zabbix.
Sprawdź pliki dziennika, aby upewnić się, że proxy zostało pomyślnie uruchomione.

Proxy Zabbix automatycznie zaktualizuje bazę danych.
Aktualizacja bazy danych przebiega podobnie jak podczas uruchamiania serwera Zabbix.

Proces aktualizacji agent

> **Attention:**
> Aktualizacja agentów nie jest obowiązkowa. Agenty należy aktualizować tylko wtedy, gdy jest to wymagane do uzyskania dostępu do nowych funkcji.

Procedura aktualizacji opisana w tej sekcji może być używana zarówno do aktualizacji Zabbix agent, jak i Zabbix agent 2.

1 Zatrzymaj agent

Zatrzymaj agent Zabbix.

2 Utwórz kopię zapasową plików konfiguracyjnych i plików binarnych agenta Zabbix

Utwórz kopię zapasową pliku binarnego agenta Zabbix oraz pliku konfiguracyjnego.

3 Instalowanie nowych plików binarnych agentów

Skorzystaj z tych instrukcji, aby skompilować agenta Zabbix ze źródeł.

Alternatywnie możesz pobrać wstępnie skompilowane agenty Zabbix ze strony pobierania Zabbix.

4 Przejrzyj parametry konfiguracji agenta

Upewnij się, że zapoznałeś się z Uwagi dotyczące aktualizacji, aby sprawdzić, czy wymagane są jakiekolwiek zmiany w parametrach konfiguracji.

Informacje o nowych opcjonalnych parametrach znajdziesz na stronie Co nowego.

5 Uruchom nowy agent Zabbix

Uruchom nowy agent Zabbix. Sprawdź pliki dziennika, aby upewnić się, że agent został pomyślnie uruchomiony.

Aktualizacja między wersjami pomniejszymi

Podczas aktualizacji między wersjami pomniejszymi 8.0.x (na przykład z 8.0.1 do 8.0.3) należy wykonać te same czynności dla serwera/proxy/agenta co podczas aktualizacji między wersjami głównymi. Jedyna różnica polega na tym, że podczas aktualizacji między wersjami pomniejszymi nie są wprowadzane żadne zmiany w bazie danych.

Znane problemy

Zobacz także: Problemy z kompilacją.

Aktualizacja

Ustawienie trybu SQL dla pomyślnej aktualizacji

Ustawienie `sql_mode` w MySQL/MariaDB musi mieć ustawiony tryb "STRICT_TRANS_TABLES". Jeśli go brakuje, aktualizacja bazy danych Zabbix zakończy się niepowodzeniem (zobacz także ZBX-19435).

Aktualizacja z MariaDB 10.2.1 i wcześniejszą

Aktualizacja Zabbix może się nie powieść, jeśli tabele bazy danych zostały utworzone przy użyciu MariaDB 10.2.1 lub wcześniejszej, ponieważ w tych wersjach domyślny format wiersza to compact. Można to naprawić, zmieniając format wiersza na dynamic (zobacz także ZBX-17690).

Zestaw znaków i sortowanie bazy danych MySQL

Instalacje Zabbix korzystające z bazy danych MySQL utworzonej z zestawem znaków utf8mb3 i sortowaniem utf8mb3_bin mogą napotkać problemy. Na przykład znany problem MySQL może powodować błędy `Row size too large (> 8126)` podczas aktualizacji schematu bazy danych lub wykonywania innych operacji DDL.
 Zdecydowanie zaleca się przekonwertowanie baz danych korzystających z zestawu znaków utf8mb3 i sortowania utf8mb3_bin na zestaw znaków utf8mb4 i sortowanie utf8mb4_bin, zgodnie z opisem w sekcji Naprawianie zestawu znaków i sortowania bazy danych Zabbix.

Wyciek pamięci i deskryptorów plików w libcurl 8.21.0

Znany problem w libcurl 8.21.0 może powodować wycieki pamięci i deskryptorów plików w funkcjach Zabbix korzystających z libcurl podczas obsługi połączeń zabezpieczonych TLS.
 Obejściem tego problemu jest obniżenie wersji libcurl do 8.20.0 do czasu udostępnienia wydania zawierającego poprawkę.

Szablony

Zgodność szablonów w środowiskach dual-stack (IPv4/IPv6)

W środowiskach dual-stack (systemach skonfigurowanych do obsługi zarówno IPv4, jak i IPv6) nazwa hosta `localhost` zwykle jest rozwiązywana zarówno do adresów IPv4, jak i IPv6. Ze względu na częste priorytetyzowanie IPv6 względem IPv4 przez wiele systemów operacyjnych i resolverów DNS, szablony Zabbix mogą nie działać poprawnie, jeśli monitorowana usługa jest skonfigurowana do nasłuchiwania wyłącznie na IPv4.

Usługi, które nie są skonfigurowane do nasłuchiwania na adresach IPv6, mogą stać się niedostępne, co prowadzi do niepowodzeń monitorowania. Użytkownicy mogą poprawnie skonfigurować dostęp dla IPv4, a mimo to nadal napotykać problemy z łącznością z powodu domyślnego zachowania polegającego na priorytetyzowaniu IPv6.

Obejściem tego problemu jest upewnienie się, że usługi (Nginx, Apache, PostgreSQL itp.) są skonfigurowane do nasłuchiwania zarówno na adresach IPv4, jak i IPv6, a serwer/agent Zabbix ma dozwolony dostęp przez IPv6. Dodatkowo w szablonach i konfiguracjach Zabbix należy jawnie używać `localhost` zamiast `127.0.0.1`, aby zapewnić zgodność zarówno z IPv4, jak i IPv6.

Na przykład, podczas monitorowania PostgreSQL za pomocą szablonu PostgreSQL by Zabbix agent 2, może być konieczna edycja pliku `pg_hba.conf`, aby zezwolić na połączenia dla użytkownika `zbx_monitor`. Jeśli środowisko dual-stack nadaje priorytet IPv6 (system rozwiązuje localhost do `::1`) i skonfigurujesz `localhost`, ale dodasz tylko wpis IPv4 (`127.0.0.1/32`), połączenie zakończy się niepowodzeniem, ponieważ nie ma pasującego wpisu IPv6.

Poniższy przykład pliku `pg_hba.conf` zapewnia, że użytkownik `zbx_monitor` może łączyć się z dowolną bazą danych z lokalnej maszyny przy użyciu zarówno adresów IPv4, jak i IPv6, z różnymi metodami uwierzytelniania:

```
### 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
```

W razie potrzeby można również użyć bezpośrednio adresu IPv4 (`127.0.0.1`) podczas konfigurowania makra szablonu PostgreSQL by Zabbix agent 2 dla ciągu połączenia.

Przypadkowa instalacja pakietów Zabbix z EPEL

Jeśli repozytorium EPEL jest zainstalowane i włączone, instalacja pakietów Zabbix może spowodować pobranie wersji z EPEL zamiast oficjalnych pakietów Zabbix. Aby rozwiązać ten problem:

1. Usuń wszystkie pakiety Zabbix zainstalowane z EPEL:

```
dnf remove zabbix-server-mysql
```

2. Wyklucz pakiety Zabbix z EPEL, dodając następujący wiersz do pliku `/etc/yum.repos.d/epel.repo`:

```
[epel]
...
excludepkgs=zabbix*
```

3. Zainstaluj ponownie oficjalny pakiet serwera Zabbix:

```
dnf install zabbix-server-mysql
```

Podczas instalacji oficjalne pakiety Zabbix zawierają słowo `release` w ciągu wersji (na przykład `7.0.0-release1.el8`), co odróżnia je od pakietów EPEL.

Pakiety Zabbix dla RHEL w środowiskach Red Hat UBI

Podczas instalacji Zabbix z pakietów Red Hat Enterprise Linux w środowiskach Red Hat Universal Base Image należy zapewnić dostęp do wymaganych repozytoriów i zależności. Pakiety Zabbix zależą od bibliotek `libOpenIPMI.so` i `libOpenIPMIposix.so`, które nie są dostarczane przez żaden pakiet w domyślnych repozytoriach menedżera pakietów włączonych w systemach UBI, co spowoduje niepowodzenie instalacji.

Biblioteki `libOpenIPMI.so` i `libOpenIPMIposix.so` są dostępne w pakiecie `OpenIPMI-libs`, który jest udostępniany przez repozytorium `redhat-#-for-<arch>-appstream-rpms`. Dostęp do tego repozytorium jest regulowany przez subskrypcje, które w przypadku środowisk UBI są propagowane przez montowanie katalogów konfiguracji repozytoriów i sekretów hosta RHEL do przestrzeni nazw systemu plików kontenera.

Aby uzyskać więcej informacji, zobacz ZBX-24291.

Wygasły klucz podpisujący dla pakietów RHEL

Podczas aktualizacji Zabbix w systemie Red Hat Enterprise Linux lub jego pochodnych możesz napotkać problem z wygasłym kluczem podpisującym pakiety w repozytorium Zabbix. Gdy klucz podpisujący wygaśnie, próby weryfikacji podpisów pakietów zakończą się błędem wskazującym, że certyfikat lub klucz nie jest już ważny. Na przykład:

```
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
```

Aby rozwiązać taki problem, ręcznie zainstaluj ponownie najnowszy pakiet `zabbix-release` dla konkretnego wariantu RHEL (zastąp poniższy link właściwym linkiem z repozytorium Zabbix).

Na przykład w systemie **RHEL 10** uruchom:

```
rpm -Uvh https://repo.zabbix.com/zabbix/8.0/release/rhel/10/noarch/zabbix-release-latest.el10.noarch.rpm
```

Następnie zaktualizuj informacje o repozytorium:

```
dnf update
```

Więcej informacji znajdziesz w ZBX-24761.

Połączenie TLS z bazą danych z MariaDB

Połączenie TLS z bazą danych nie jest obsługiwane z opcją `verify_ca` dla parametru DBTLSConnect, jeśli używana jest MariaDB.

Możliwe zakleszczenia z MySQL/MariaDB

Podczas pracy pod dużym obciążeniem, gdy zaangażowany jest więcej niż jeden worker LLD, może dojść do zakleszczenia spowodowanego błędem InnoDB związanym ze strategią blokowania wierszy (zobacz błąd upstream). Błąd został naprawiony w MySQL od wersji 8.0.29, ale nie w MariaDB. Więcej informacji można znaleźć w ZBX-21506.

Globalna korelacja zdarzeń

Zdarzenia mogą nie zostać skorelowane poprawnie, jeśli odstęp czasu między pierwszym a drugim zdarzeniem jest bardzo mały, tj. pół sekundy lub mniej.

NetBSD 8.0 i nowsze

Różne procesy Zabbix mogą losowo ulegać awarii podczas uruchamiania w wersjach NetBSD 8.X i 9.X. Jest to spowodowane zbyt małym domyślnym rozmiarem stosu (4 MB), który należy zwiększyć, uruchamiając:

```
ulimit -s 10240
```

Aby uzyskać więcej informacji, zobacz powiązane zgłoszenie problemu: ZBX-18275.

Ograniczenia wyrażeń regularnych w Zabbix agent 2

Zabbix agent 2 nie obsługuje asercji lookahead i lookbehind w wyrażeniach regularnych ze względu na ograniczenia standardowej biblioteki Go regexp.

Kontrole IPMI

Kontrole IPMI nie będą działać ze standardowym pakietem biblioteki OpenIPMI w Debianie wcześniejszym niż 9 (stretch) oraz Ubuntu wcześniejszym niż 16.04 (xenial). Aby to naprawić, należy ponownie skompilować bibliotekę OpenIPMI z włączoną obsługą OpenSSL, jak omówiono w ZBX-6139.

IPMI — niezaufane hosty mogą powodować awarię OpenIPMI

W bibliotece OpenIPMI używanej przez Zabbix do odpytywania danych IPMI występuje błąd, który może zostać wywołany przez specjalnie spreparowane odpowiedzi z niezaufanego urządzenia. Niezaufane urządzenie IPMI może wysłać spreparowane dane powodujące awarię biblioteki OpenIPMI, co z kolei może doprowadzić do zakończenia procesu serwera Zabbix wykonującego odpytywanie IPMI.

Kontrole SSH

- Niektóre dystrybucje Linuksa, takie jak Debian i Ubuntu, nie obsługują zaszyfrowanych kluczy prywatnych (z hasłem) w przypadku, gdy biblioteka libssh2 jest zainstalowana z pakietów. Więcej informacji znajduje się w ZBX-4850.
- W przypadku używania libssh 0.9.x w niektórych dystrybucjach Linuksa z OpenSSH 8, kontrole SSH mogą sporadycznie zgłaszać błąd „Cannot read data from SSH server". Jest to spowodowane problemem w libssh (bardziej szczegółowy raport). Oczekuje się, że błąd został naprawiony w stabilnym wydaniu libssh 0.9.5. Szczegóły znajdują się także w ZBX-17756.
- Użycie znaku potoku „|" w skrypcie SSH może prowadzić do błędu „Cannot read data from SSH server". W takim przypadku zaleca się aktualizację wersji biblioteki libssh. Szczegóły znajdują się także w ZBX-21337.

Kontrole ODBC

- Sterownik MySQL unixODBC nie powinien być używany z serwer Zabbix lub proxy Zabbix skompilowanymi z biblioteką łącznika MariaDB i odwrotnie; jeśli to możliwe, lepiej również unikać używania tego samego łącznika co sterownika ze względu na błąd po stronie projektu upstream. Zalecana konfiguracja:

  Łącznik PostgreSQL, SQLite lub Oracle → sterownik MariaDB lub MySQL unixODBC
Łącznik MariaDB → sterownik MariaDB unixODBC
Łącznik MySQL → sterownik MySQL unixODBC

  Więcej informacji i dostępne obejścia można znaleźć w ZBX-7665.

- Dane XML pobierane z Microsoft SQL Server mogą być obcinane na różne sposoby w systemach Linux i UNIX.
- Zaobserwowano, że używanie kontroli ODBC do monitorowania baz danych Oracle przy użyciu różnych wersji Oracle Instant Client dla systemu Linux powoduje awarie serwer Zabbix.
 Zobacz także: ZBX-18402, ZBX-20803.
- W przypadku używania sterownika FreeTDS UnixODBC należy dodać na początku zapytania SQL instrukcję 'SET NOCOUNT ON' (na przykład `SET NOCOUNT ON DECLARE @strsql NVARCHAR(max) SET @strsql = ....`). W przeciwnym razie pozycja monitorowania bazy danych w Zabbix nie będzie w stanie pobrać informacji i zwróci błąd "SQL query returned empty result".
 Więcej informacji można znaleźć w ZBX-19917.

Nieprawidłowy parametr metody żądania w pozycjach

Parametr metody żądania, używany wyłącznie w kontrolach HTTP, może być nieprawidłowo ustawiony na „1", czyli wartość inną niż domyślna dla wszystkich pozycji, w wyniku aktualizacji z wersji Zabbix starszej niż 4.0. Szczegółowe informacje o tym, jak naprawić tę sytuację, można znaleźć w ZBX-19308.

Monitorowanie WWW i agent HTTP

Serwer Zabbix powoduje wyciek pamięci w niektórych dystrybucjach Linuksa z powodu błędu upstream, gdy opcja „SSL verify peer" jest włączona w scenariuszach WWW lub w agencie HTTP. Więcej informacji oraz dostępne obejścia problemu można znaleźć w ZBX-10486.

Proste testy

W wersjach **fping** wcześniejszych niż v3.10 występuje błąd powodujący nieprawidłową obsługę zduplikowanych pakietów odpowiedzi echo. Może to powodować nieoczekiwane wyniki dla pozycji `icmpping`, `icmppingloss`, `icmppingsec`. Zaleca się używanie najnowszej wersji **fping**. Więcej szczegółów można znaleźć w ZBX-11726.

Błędy wykonywania fping w kontenerach rootless

Gdy kontenery działają w trybie rootless lub w środowisku z określonymi ograniczeniami, podczas wykonywania kontroli ICMP mogą wystąpić błędy związane z uruchamianiem fping, takie jak `fping: Operation not permitted` lub utrata wszystkich pakietów do wszystkich zasobów.

Aby rozwiązać ten problem, dodaj `--cap-add=net_raw` do poleceń "docker run" lub "podman run".

Dodatkowo uruchamianie fping w środowiskach bez uprawnień roota może wymagać modyfikacji sysctl, tj.:

```
sudo sysctl -w "net.ipv4.ping_group_range=0 1995"
```

gdzie "1995" to GID użytkownika zabbix. Więcej informacji można znaleźć w ZBX-22833.

Kontrole SNMP

Jeśli używany jest system operacyjny OpenBSD, błąd use-after-free w bibliotece Net-SNMP do wersji 5.7.3 może spowodować awarię serwera Zabbix, jeśli parametr SourceIP jest ustawiony w pliku konfiguracyjnym serwera Zabbix. Jako obejście problemu nie należy ustawiać parametru SourceIP. Ten sam problem dotyczy również systemu Linux, ale nie powoduje zatrzymania pracy serwera Zabbix. Lokalna poprawka dla pakietu net-snmp w OpenBSD została zastosowana i zostanie wydana wraz z OpenBSD 6.3.

Skoki danych SNMP

Zaobserwowano skoki danych SNMP, które mogą być związane z pewnymi czynnikami fizycznymi, takimi jak skoki napięcia w sieci elektrycznej. Więcej szczegółów znajduje się w ZBX-14318.

Pułapki SNMP

Pakiet "net-snmp-perl", wymagany do obsługi pułapek SNMP, został usunięty w RHEL 8.0-8.2; ponownie dodany w RHEL 8.3.

Jeśli więc używasz RHEL 8.0-8.2, najlepszym rozwiązaniem jest aktualizacja do RHEL 8.3.

Aby uzyskać więcej informacji, zobacz także ZBX-17192.

Awaria procesu alerter w RHEL 7

W RHEL 7 odnotowano przypadki awarii procesu alerter serwera Zabbix. Szczegółowe informacje znajdują się w ZBX-10461.

Aktualizacja Zabbix agent 2 (6.0.5 lub starszy)

Podczas aktualizacji Zabbix agent 2 (wersja 6.0.5 lub starsza) z pakietów może wystąpić błąd konfliktu plików związany z wtyczkami. Aby naprawić ten błąd, wykonaj kopię zapasową konfiguracji agent 2 (jeśli to konieczne), odinstaluj agent 2 i zainstaluj go ponownie.

W systemach opartych na RHEL uruchom:

```
dnf remove zabbix-agent2
dnf install zabbix-agent2
```

W systemach opartych na Debian uruchom:

```
apt remove zabbix-agent2
apt install zabbix-agent2
```

Aby uzyskać więcej informacji, zobacz ZBX-23250.

Zmiana ustawień regionalnych frontendu

Zaobserwowano, że ustawienia regionalne frontendu mogą zmieniać się bez wyraźnej przyczyny, tzn. niektóre strony (lub ich części) są wyświetlane w jednym języku, podczas gdy inne strony (lub ich części) w innym języku. Zazwyczaj problem może wystąpić, gdy jest kilku użytkowników, z których niektórzy używają jednych ustawień regionalnych, a inni innych.

Znanym sposobem obejścia tego problemu jest wyłączenie wielowątkowości w PHP i Apache.

Problem jest związany ze sposobem ustawiania ustawień regionalnych w PHP: informacje o ustawieniach regionalnych są przechowywane dla procesu, a nie dla wątku. W środowisku wielowątkowym, gdy ten sam proces Apache uruchamia kilka projektów, ustawienia regionalne mogą zostać zmienione w innym wątku, co może wpłynąć na sposób przetwarzania danych w wątku Zabbix.

Więcej informacji można znaleźć w powiązanych zgłoszeniach:

- ZBX-10911 (Problem ze zmianą ustawień regionalnych frontendu)
- ZBX-16297 (Problem z przetwarzaniem liczb na wykresach przy użyciu funkcji `bcdiv` z funkcji BC Math)

Wykresy

Problemy z wykresami (klasycznymi)

Jeśli napotkasz problemy z klasycznymi wykresami, zaleca się aktualizację biblioteki GD (libgd) do wersji 2.3.3-13 lub nowszej oraz PHP do wersji 8.0.19, 8.1.33, 8.2.29, 8.3.25, 8.4.12 lub nowszej.

Czas letni

Zmiany związane z czasem letnim (DST) powodują nieprawidłowości podczas wyświetlania etykiet osi X (duplikacja dat, brak dat itp.).

Agregacja sumy

W przypadku użycia agregacji sumy na wykresie dla okresu krótszego niż jedna godzina, wykresy wyświetlają nieprawidłowe (pomnożone) wartości, gdy dane pochodzą z trendów.

Nakładanie się tekstu

W przypadku niektórych języków frontendu (np. japońskiego) lokalne czcionki mogą powodować nakładanie się tekstu w legendzie wykresu. Aby tego uniknąć, użyj PHP GD extension w wersji 2.3.0 (lub nowszej).

Monitorowanie plików dziennika

Pozycje `log[]` i `logrt[]` wielokrotnie odczytują ponownie plik dziennika od początku, jeśli system plików jest zapełniony w 100% i do pliku dziennika są dopisywane nowe dane (więcej informacji można znaleźć w ZBX-10884).

Wolne zapytania MySQL

Serwer Zabbix generuje wolne zapytania `SELECT` w przypadku nieistniejących wartości dla pozycji. Ten problem występuje w wersjach MySQL 5.6/5.7 (szersze omówienie znajduje się w ZBX-10652) i w określonych przypadkach może również występować w nowszych wersjach MySQL. Obejściem tego problemu jest wyłączenie optymalizatora `index_condition_pushdown` lub `prefer_ordering_index` w MySQL. Należy jednak pamiętać, że to obejście może nie rozwiązać wszystkich problemów związanych z wolnymi zapytaniami.

Trwałe ustawienia filtra z linków

Po otwarciu linku do strony Zabbix frontend, który zawiera ustawienia filtra, w tym selektor czasu, filtr jest automatycznie zapisywany w bazie danych dla użytkownika, zastępując wcześniej zapisany filtr i/lub ustawienia selektora czasu dla tej strony. Ustawienia te pozostają aktywne, dopóki użytkownik nie zaktualizuje ich ręcznie lub nie zresetuje.

Problem z adresem IPv6 w pułapkach SNMPv3

Z powodu błędu net-snmp adres IPv6 może nie być poprawnie wyświetlany podczas używania SNMPv3 w pułapkach SNMP. Więcej szczegółów oraz możliwe obejście problemu można znaleźć w ZBX-14541.

Przycięty długi adres IPv6 w informacji o nieudanym logowaniu

Komunikat o nieudanej próbie logowania wyświetli tylko pierwsze 39 znaków zapisanego adresu IP, ponieważ jest to limit znaków w polu bazy danych. Oznacza to, że adresy IPv6 dłuższe niż 39 znaków będą wyświetlane niekompletnie.

Kontrole Zabbix agent na Windows

Nieistniejące wpisy DNS w parametrze `Server` pliku konfiguracyjnego Zabbix agent (`zabbix_agentd.conf`) mogą zwiększać czas odpowiedzi Zabbix agent w systemie Windows. Dzieje się tak, ponieważ demon buforowania DNS w systemie Windows nie buforuje negatywnych odpowiedzi dla adresów IPv4. Jednak w przypadku adresów IPv6 negatywne odpowiedzi są buforowane, więc możliwym obejściem tego problemu jest wyłączenie IPv4 na host.

Eksport/import YAML

Istnieją pewne znane problemy z YAML eksportem/importem:

- Komunikaty o błędach nie są tłumaczalne;
- Prawidłowy JSON z rozszerzeniem pliku .yaml czasami nie może zostać zaimportowany;
- Nieujęte w cudzysłów daty czytelne dla człowieka są automatycznie konwertowane na znaczniki czasu Unix.

Kreator konfiguracji na SUSE z NGINX i php-fpm

Kreator konfiguracji frontend nie może zapisać pliku konfiguracyjnego w SUSE z NGINX + php-fpm. Jest to spowodowane ustawieniem w jednostce /usr/lib/systemd/system/php-fpm.service, które uniemożliwia Zabbixowi zapisywanie do /etc. (wprowadzone w PHP 7.4).

Dostępne są dwa obejścia:

- Ustaw opcję ProtectSystem na 'true' zamiast 'full' w jednostce systemd php-fpm.
- Ręcznie zapisz plik `/etc/zabbix/web/zabbix.conf.php`.

Przekazywanie nagłówka Authorization

W niektórych przypadkach Apache lub NGINX mogą uniemożliwiać dotarcie nagłówka Authorization w żądaniach API do Zabbix. Może to powodować problemy z uwierzytelnianiem podczas korzystania z Zabbix API lub usług jednokrotnego logowania (SSO), takich jak SAML z Okta.

Aby rozwiązać ten problem, zaktualizuj konfigurację swojego serwera WWW.

W przypadku **Apache**, jeśli używasz go jako reverse proxy (konfiguracja bez CGI), dodaj następującą dyrektywę do `/etc/httpd/conf/httpd` (w systemach opartych na RHEL) lub `/etc/apache2/apache2.conf` (w Debian/Ubuntu):

```
SetEnvIfNoCase ^Authorization$ "(.+)" HTTP_AUTHORIZATION=$1
```

Jeśli Apache bezpośrednio wykonuje skrypty obsługujące żądania (np. przy użyciu mod_cgi), zamiast tego dodaj następującą dyrektywę:

```
CGIPassAuth On
```

Natomiast **NGINX** obsługuje nagłówek Authorization automatycznie. Jeśli jednak NGINX działa jako reverse proxy, możesz jawnie przekazywać nagłówek Authorization, dodając następujące dyrektywy do `/etc/nginx/nginx.conf` (dla lokalizacji twojego frontend Zabbix):

```
...
location / {
...
    proxy_set_header Authorization $http_authorization;
    proxy_pass http://backend_server;
...
}
```

Po zaktualizowaniu konfiguracji uruchom ponownie swój serwer WWW.

Więcej informacji znajdziesz tutaj:

- ZBX-22952
- Apache 2.4 + PHP-FPM and Authorization headers
- Dyrektywy SetEnvIfNoCase i CGIPassAuth
- NGINX Reverse Proxy

Chromium dla usługi web Zabbix w Ubuntu 20

Chociaż w większości przypadków usługa web Zabbix może działać z Chromium, w Ubuntu 20.04 użycie Chromium powoduje następujący błąd:

```
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
```

Ten błąd występuje, ponieważ `/var/lib/zabbix` jest używany jako katalog domowy użytkownika „zabbix”.

Niestandardowe kody błędów MySQL

Gdy Zabbix wykryje, że baza danych backend jest niedostępna, wysyła powiadomienie i nadal próbuje się połączyć. Dla niektórych silników baz danych rozpoznawane są określone kody błędów. W MySQL do rozpoznawanych kodów błędów należą:

- 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

Dodatkowo podczas używania Zabbix z instalacją MySQL na platformie Azure w logach Zabbix może pojawić się ogólny komunikat o błędzie *[9002] Some errors occurred*. Ten komunikat jest wysyłany przez bazę danych do serwera lub proxy Zabbix. Aby ustalić przyczynę błędu, sprawdź logi Azure.

Nieprawidłowe wyrażenia regularne po przełączeniu na PCRE2

W Zabbix 6.0 dodano obsługę PCRE2. Mimo że PCRE jest nadal obsługiwane, pakiety instalacyjne Zabbix dla RHEL 7 i nowszych, SLES (wszystkie wersje), Debian 9 i nowszych oraz Ubuntu 16.04 i nowszych zostały zaktualizowane tak, aby używać PCRE2. Choć przynosi to wiele korzyści, przełączenie na PCRE2 może spowodować, że niektóre istniejące wzorce wyrażeń regularnych PCRE staną się nieprawidłowe lub będą działać inaczej. W szczególności dotyczy to wzorca

[\w-\.]. Aby ponownie uczynić to wyrażenie regularne poprawnym bez wpływu na jego semantykę, zmień wyrażenie na *[-\w\.]* . Dzieje się tak dlatego, że PCRE2 traktuje znak myślnika jako separator, tworząc zakres wewnątrz klasy znaków.

Błąd widżetu Geomap

Mapy w widżecie Geomap mogą nie ładować się poprawnie, jeśli dokonano aktualizacji z wcześniejszej wersji Zabbix z NGINX i nie przełączono się na nowy plik konfiguracyjny NGINX podczas aktualizacji.

Aby rozwiązać problem, możesz odrzucić stary plik konfiguracyjny, użyć pliku konfiguracyjnego z pakietu bieżącej wersji i ponownie go skonfigurować zgodnie z opisem w instrukcji pobierania w sekcji *e. Configure PHP for Zabbix frontend*.

Alternatywnie możesz ręcznie edytować istniejący plik konfiguracyjny NGINX (zwykle `/etc/zabbix/nginx.conf`). W tym celu otwórz plik i znajdź następujący blok:

```
location ~ /(api\/|conf[^\.]|include|locale|vendor) {
        deny            all;
        return          404;
}
```

Następnie zastąp ten blok następującym:

```
location ~ /(api\/|conf[^\.]|include|locale) {
        deny            all;
        return          404;
}

location /vendor {
        deny            all;
        return          404;
}
```

Wstępne przetwarzanie — zmienne globalne są niebezpieczne

JavaScript we wstępnym przetwarzaniu jest uruchamiany dla każdego żądania, ale przypisania do niezadeklarowanych identyfikatorów (na przykład `secret = value`) tworzą niejawne zmienne globalne, które mogą utrzymywać się poza bieżącym wykonaniem. Przechowywanie danych wrażliwych (tokenów, haseł itp.) w niejawnych zmiennych globalnych zwiększa ryzyko przypadkowego ujawnienia lub ponownego użycia przez kolejne uruchomienia wstępnego przetwarzania albo inne integracje wykonywane w tym samym środowisku.

Nie polegaj na niejawnych zmiennych globalnych. Zawsze deklaruj zmienne za pomocą `var` lub `const` i unikaj dołączania sekretów do obiektów globalnych (na przykład `globalThis` lub `window`). Nie ma obsługiwanego sposobu zastępowania wbudowanych obiektów globalnych z poziomu wstępnego przetwarzania.

Bezpieczny przykład:

```
var apiToken = payload.token;
var count = 1;
return JSON.stringify({ token: apiToken, calls: count });
```

Grupy procesorów w systemie Windows

Dokumentacja firmy Microsoft stwierdza, że systemy z mniej niż 64 procesorami logicznymi zawsze mają jedną grupę procesorów, Grupę 0. Jednak użytkownicy Zabbix zgłosili rzadki błąd ZBX-20260, gdy w systemach z 64 lub mniejszą liczbą procesorów logicznych występowały dwie grupy procesorów. Skutkowało to dostępnością liczników wydajności "\Processor(n)" tylko dla jednej z dwóch grup procesorów. Rzeczywista przyczyna źródłowa tego błędu nie jest znana. Jednak podobny przypadek opisano na stackoverflow.com, a jego przyczyną była współpraca między BIOS-em a systemem Windows.

Ograniczenia filtrowania przy użyciu porównań utf8mb4

Filtry (np. w *Data collection* > *Maintenance*) mogą nie działać poprawnie, gdy są stosowane do encji zawierających określone znaki Unicode (np. ℰ, ∉). Problem ten wynika ze sposobu, w jaki domyślne porównanie utf8mb4_bin w bazach danych MySQL lub MariaDB obsługuje sortowanie i porównywanie znaków Unicode.

Aby obejść to ograniczenie, użytkownicy mogą zmienić porównanie kolumn bazy danych na alternatywne, takie jak utf8mb4_0900_bin, utf8mb4_0900_ai_ci lub utf8mb4_unicode_520_ci. Należy jednak pamiętać, że zmiana porównania może powodować nieoczekiwane zachowanie przy obsłudze pustych znaków, a także podczas sortowania i filtrowania innych znaków.

Więcej informacji o zmianie porównań można znaleźć w dokumentacji MySQL lub dokumentacji MariaDB. Szczegółowe informacje o różnicach między porównaniami znajdują się w sekcji Unicode Character Sets w dokumentacji MySQL.

Profilowanie nadmiernego użycia pamięci za pomocą tcmalloc

Jeśli podejrzewasz, że Twoja instalacja Zabbix zużywa zbyt dużo pamięci, możesz użyć funkcji profilowania pamięci tcmalloc, aby zbadać zużycie pamięci przez serwer/proxy Zabbix.

1. Podczas instalacji Zabbix ze źródeł skonfiguruj dodatkowe flagi:

```
export CFLAGS="-std=gnu99 -g -O0"
```

Flaga `-std=gnu99` jest wymagana do budowania serwera Zabbix, proxy Zabbix lub agent Zabbix. Flaga `-g` dodaje dodatkowe informacje debugowania, natomiast `-O0` wyłącza optymalizacje, które mogą zakłócać profilowanie tcmalloc.

2. Ustaw następujące zmienne środowiskowe przed uruchomieniem serwera Zabbix. Zmienne te informują tcmalloc, jak śledzić i raportować użycie pamięci:

```
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. Wygeneruj zrzut profilu, wysyłając sygnał 5 do procesu docelowego. Zastąp 1234 rzeczywistym identyfikatorem procesu (PID):

```
kill -5 1234
```

4. Wyświetl wygenerowany profil:

```
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
```

W tym przykładzie zbx_malloc2 odpowiada za niemal wszystkie alokacje pamięci.

Zobacz także:

- ZBX-25050 oraz ZBX-25584 — powiązane zgłoszenia problemów.
- GCC Option Summary — opcje kompilacji (`-std=gnu99`, `-g`, `-O0` itd.).
- Dokumentacja Gperftools Heap Profiler dotycząca zmiennych środowiskowych dla profilowania tcmalloc.

Replikacja grupowa MySQL 8.0 w trybie multi-primary

Podczas korzystania z replikacji grupowej MySQL 8.0 w trybie multi-primary możesz napotkać błąd podczas zatwierdzania transakcji podobny do poniższego:

```
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
```

Wygląda na to, że ten błąd jest wywoływany przez problemy z operacjami wycofywania transakcji obejmującymi ograniczenia kluczy obcych.

Zobacz także:

- ZBX-26060 — powiązane zgłoszenie problemu.
- MySQL Bug #96758 "Rollbacks with Foreign Keys on single node" — problem zgłoszony po stronie upstream.

Logowanie i systemd

Zalecamy utworzenie użytkownika *zabbix* jako użytkownika systemowego, czyli bez możliwości logowania. Niektórzy użytkownicy ignorują to zalecenie i używają tego samego konta do logowania (np. za pomocą SSH) do hosta, na którym działa Zabbix. Może to spowodować awarię demona Zabbix podczas wylogowywania. W takim przypadku w logu serwera Zabbix pojawi się coś podobnego do poniższego:

```
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
```

a w logu agenta Zabbix:

```
zabbix_agentd [27796]: [file:'log.c',line:238] lock failed: [22] Invalid argument
```

Dzieje się tak z powodu domyślnego ustawienia systemd `RemoveIPC=yes`, skonfigurowanego w pliku `/etc/systemd/logind.conf`. Podczas wylogowywania się z systemu semafory utworzone wcześniej przez Zabbix są usuwane, co powoduje awarię.

Cytat z dokumentacji 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".
```

Istnieją 2 rozwiązania tego problemu:

1. (zalecane) Nie używaj konta *zabbix* do celów innych niż procesy Zabbix; utwórz osobne konto do pozostałych zastosowań.
2. (niezalecane) Ustaw `RemoveIPC=no` w pliku `/etc/systemd/logind.conf` i uruchom ponownie system. Pamiętaj, że `RemoveIPC` jest parametrem obowiązującym w całym systemie, więc jego zmiana wpłynie na cały system.

Wymagania

Sprzęt

Pamięć

Zabbix wymaga zarówno pamięci fizycznej, jak i pamięci dyskowej. Wymagana ilość pamięci dyskowej zależy oczywiście od liczby hostów i parametrów, które są monitorowane. Jeśli planujesz przechowywać długą historię monitorowanych parametrów, powinieneś przewidzieć co najmniej kilka gigabajtów, aby mieć wystarczająco dużo miejsca na zapisanie historii w bazie danych. Każdy proces demona Zabbix wymaga kilku połączeń z serwerem bazy danych. Ilość pamięci przydzielonej dla połączenia zależy od konfiguracji silnika bazy danych.

> **Note:**
> Im więcej masz pamięci fizycznej, tym szybciej działa baza danych (a tym samym Zabbix).

CPU

Zabbix, a w szczególności baza danych Zabbix, może wymagać znacznych zasobów CPU w zależności od liczby monitorowanych parametrów i wybranego silnika bazy danych.

Inny sprzęt

Port komunikacji szeregowej oraz szeregowy modem GSM są wymagane do korzystania z obsługi powiadomień SMS w Zabbix. Konwerter USB-do-portu-szeregowego również będzie działać.

Przykłady konfiguracji sprzętowej

Tabela przedstawia przykłady konfiguracji sprzętowej przy założeniu platformy **Linux/BSD/Unix**.

Są to przykłady rozmiaru i konfiguracji sprzętowej, od których można rozpocząć. Każda instalacja Zabbix jest unikalna. Pamiętaj, aby przeprowadzić testy wydajności systemu Zabbix w środowisku przejściowym lub deweloperskim, tak aby w pełni zrozumieć wymagania przed wdrożeniem instalacji Zabbix w środowisku produkcyjnym.

Rozmiar instalacji	Monitorowane metryki[1]	Rdzenie CPU/vCPU	Pamięć (GiB)	Baza danych	Amazon EC2[2]
Mała	1 000	2	8	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.large/m6g.large
Średnia	10 000	4	16	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.xlarge/m6g.xlarge
Duża	100 000	16	64	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.4xlarge/m6g.4xlarge
Bardzo duża	1 000 000	32	96	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.8xlarge/m6g.8xlarge

[1] Na potrzeby określania rozmiaru 1 monitorowana metryka jest w przybliżeniu reprezentowana przez wzór: 1 metryka = 1 pozycja + 1 wyzwalacz + 1 wykres. Jest to wyłącznie reguła szacunkowa, a nie wzór do obliczania liczby monitorowanych metryk.
[2] Przykład z ogólnego przeznaczenia instancji Amazon EC2, korzystających z architektury ARM64 lub x86_64. Podczas oceny i testowania instalacji Zabbix, przed zainstalowaniem jej w środowisku produkcyjnym, należy wybrać odpowiedni typ instancji, na przykład zoptymalizowany pod kątem obliczeń, pamięci lub pamięci masowej.

Note:
Rzeczywista konfiguracja w dużym stopniu zależy od liczby aktywnych pozycji i częstotliwości odświeżania (szczegółowe informacje zawiera sekcja Rozmiar bazy danych na tej stronie). Zdecydowanie zaleca się uruchomienie bazy danych na osobnym serwerze w przypadku dużych instalacji.

Obsługiwane platformy

Ze względu na wymagania bezpieczeństwa oraz krytyczny charakter serwera monitoringu, UNIX jest jedynym systemem operacyjnym, który może stale zapewniać wymaganą wydajność, odporność na awarie i niezawodność. Zabbix działa na wiodących na rynku wersjach systemów.

Komponenty Zabbix są dostępne i przetestowane dla następujących platform:

Platforma	Serwer	Agent	Agent 2	Uwagi
Linux	x	x	x	
Windows	-	x	x	Agent Zabbix jest obsługiwany we wszystkich desktopowych i serwerowych wersjach systemu od Windows XP (64-bit)/Server 2003. Agent Zabbix 2 jest obsługiwany we wszystkich desktopowych i serwerowych wersjach systemu od Windows 10 (32-bit)/Server 2016, ponieważ jest kompilowany wyłącznie przy użyciu obsługiwanej wersji Go, aby zapobiec krytycznym lukom bezpieczeństwa. Od Go 1.21 minimalne wymagane wersje Windows zostały podniesione, co sprawia, że Windows 10/Server 2016 jest minimalną wersją dla agenta Zabbix 2.
macOS	x	x	-	
IBM AIX	x	x	-	Agent Zabbix nie działa na platformach AIX w wersjach niższych niż 6.1 TL07 / 7.1 TL01.
FreeBSD	x	x	-	
OpenBSD	x	x	-	
Solaris	x	x	-	
NetBSD	x	x	-	
HP-UX	x	x	-	

Note:
Serwer/agent Zabbix może działać na innych systemach operacyjnych typu Unix.

Attention:
Zabbix wyłącza zrzuty core, jeśli został skompilowany z obsługą szyfrowania, i nie uruchamia się, jeśli system nie pozwala na wyłączenie zrzutów core.

Wymagane oprogramowanie

Zabbix jest oparty na nowoczesnych serwerach WWW, wiodących silnikach baz danych oraz języku skryptowym PHP.

Zewnętrzne oprogramowanie towarzyszące innych firm

Jeśli oprogramowanie lub biblioteka są oznaczone jako wymagane, ich użycie jest bezwzględnie konieczne. Opcjonalne elementy są potrzebne do obsługi określonych funkcji.

Oprogramowanie	Status wymagania	Obsługiwane wersje	Uwagi
MySQL/Percona	Jedno z	8.4.0-9.7.X	Wymagane, jeśli MySQL (lub Percona) jest używany jako baza danych backendu Zabbix. Wymagany jest silnik InnoDB. Zalecamy używanie biblioteki C API (libmysqlclient) do budowania serwera/proxy. Zdecydowanie zalecamy używanie w środowiskach produkcyjnych wyłącznie wydań LTS. Nie zaleca się wydań oznaczonych jako „rolling release”, „early access” lub podobnie.
MariaDB		10.11.00-12.3.X	Wymagany jest silnik InnoDB. Zalecamy używanie biblioteki MariaDB Connector/C do budowania serwera/proxy. Zobacz także: Możliwe zakleszczenia w MariaDB oraz Dostęp do elementów interfejsu użytkownika w MariaDB 10.5.1-10.5.9. Zdecydowanie zalecamy używanie w środowiskach produkcyjnych wyłącznie wydań LTS. Nie zaleca się wydań oznaczonych jako „rolling release”, „early access” lub podobnie.
PostgreSQL		15.0-18.X	Wymagane, jeśli PostgreSQL jest używany jako baza danych backendu Zabbix. W zależności od rozmiaru instalacji może być konieczne zwiększenie właściwości konfiguracyjnej PostgreSQL *work_mem* (wartość domyślna to 4 MB), aby ilość pamięci używanej przez bazę danych dla określonej operacji była wystarczająca, a wykonanie zapytania nie trwało zbyt długo.
TimescaleDB dla PostgreSQL		2.20.X-2.29.X	Wymagane, jeśli TimescaleDB jest używany jako rozszerzenie bazy danych PostgreSQL. Należy zainstalować TimescaleDB Community Edition, która obsługuje kompresję. Szczegółowe informacje dotyczące zgodności wersji PostgreSQL i TimescaleDB można znaleźć w dokumentacji TimescaleDB.
Elasticsearch		7.X	Wymagane, jeśli Elasticsearch jest używany jako baza danych backendu Zabbix. Elasticsearch może być używany wyłącznie do przechowywania danych historycznych i jest obsługiwany tylko przez serwery Zabbix. Obsługa Elasticsearch jest obecnie eksperymentalna. Zobacz także wymagane oprogramowanie dla serwera/proxy.

Oprogramowanie	Status wymagania	Obsługiwane wersje	Uwagi
ClickHouse		26.3.0.0-26.9.X.X	Wymagane, jeśli ClickHouse jest używany jako baza danych backendu Zabbix do przechowywania danych historycznych (obsługiwane tylko przez serwery Zabbix) lub jako źródło danych telemetrycznych. Aby używać ClickHouse jako źródła danych telemetrycznych, wymagana jest wersja ClickHouse 26.8.17.4 lub nowsza. Czas systemowy na serwerze Zabbix i serwerze ClickHouse musi być również zsynchronizowany; w przeciwnym razie zapytania telemetryczne mogą pomijać dane lub zwracać te same dane dwukrotnie.
SQLite	Opcjonalne	3.3.5-3.53.X	SQLite jest obsługiwany wyłącznie przez proxy Zabbix. Wymagane, jeśli SQLite jest używany jako baza danych proxy Zabbix.
smartmontools		7.1 lub nowsza	Wymagane dla agent 2 Zabbix.
who			Wymagane dla wtyczki zliczania użytkowników.
dpkg			Wymagane dla wtyczki system.sw.packages.
pkgtool			Wymagane dla wtyczki system.sw.packages.
rpm			Wymagane dla wtyczki system.sw.packages.
pacman			Wymagane dla wtyczki system.sw.packages.
q applets			`qlist` i `qsize`, będące częścią q applets, są wymagane dla wtyczki system.sw.packages w systemie Gentoo Linux.

Note:
Chociaż Zabbix może współpracować z bazami danych dostępnymi w systemach operacyjnych, w celu uzyskania najlepszych rezultatów zalecamy używanie baz danych zainstalowanych z oficjalnych repozytoriów twórców baz danych.

Frontend

Minimalna obsługiwana szerokość ekranu dla frontend Zabbix wynosi 1200px.

Jeśli oprogramowanie/biblioteka jest oznaczone jako wymagane, jest ono bezwzględnie niezbędne. Opcjonalne elementy są potrzebne do obsługi określonych funkcji.

Oprogramowanie	Status wymagania	Obsługiwane wersje	Uwagi
PHP	Tak	8.2.0 - 8.5.X	
Apache	Jedno z	2.4 lub nowsza	
Nginx		1.20 lub nowsza	
MySQL	Jedno z	Zobacz Oprogramowanie zewnętrzne innych firm	
PostgreSQL			
Rozszerzenia PHP			
mysqli	Tak		Wymagane, jeśli MySQL jest używany jako baza danych backendu Zabbix.
pgsql			Wymagane, jeśli PostgreSQL jest używany jako baza danych backendu Zabbix.
bcmath			php-bcmath (*--enable-bcmath*)
mbstring			php-mbstring (*--enable-mbstring*)
sockets			php-net-socket (*--enable-sockets*); wymagane do obsługi skryptów użytkownika.

Oprogramowanie	Status wymagania	Obsługiwane wersje	Uwagi
gd		2.0.28 lub nowsza	php-gd (jeśli jest dostarczane przez dystrybutora jako osobny pakiet); rozszerzenie PHP GD musi obsługiwać obrazy PNG (*--with-png-dir*), obrazy JPEG (*--with-jpeg-dir*) oraz FreeType 2 (*--with-freetype-dir*). W przypadku niektórych języków frontend może być wymagana wersja 2.3.0 lub nowsza, aby uniknąć możliwego nakładania się tekstu na wykresach.
libxml		2.6.15 lub nowsza	php-xml (jeśli jest dostarczane przez dystrybutora jako osobny pakiet)
xmlwriter			php-xmlwriter (jeśli jest dostarczane przez dystrybutora jako osobny pakiet)
xmlreader			php-xmlreader (jeśli jest dostarczane przez dystrybutora jako osobny pakiet)
ctype			php-ctype (*--enable-ctype*)
session			php-session (jeśli jest dostarczane przez dystrybutora jako osobny pakiet)
ldap	Nie		php-ldap; wymagane do uwierzytelniania LDAP.
openssl			php-openssl; wymagane do uwierzytelniania SAML.
gettext			php-gettext (*--with-gettext*); wymagane do tłumaczeń.
cURL		7.19.4 lub nowsza	php-curl; wymagane dla funkcji Duo Universal Prompt MFA, uwierzytelniania SMTP oraz testowania źródła danych telemetrycznych.

Biblioteki frontend innych firm dostarczane wraz z Zabbix:

Biblioteka	Status wymagania	Dostarczana wersja	Uwagi
jQuery JavaScript Library	Tak	3.6.0	Biblioteka JavaScript upraszczająca proces tworzenia aplikacji zgodnych z różnymi przeglądarkami.
jQuery UI		1.12.1	Zestaw interakcji z interfejsem użytkownika, efektów, widżetów i motywów zbudowanych na bazie jQuery.
SAML PHP Toolkit		4.3.1	Zestaw narzędzi PHP dodający obsługę uwierzytelniania SAML 2.0, umożliwiając logowanie do Zabbix.
Duo Universal PHP library		1.1.2	Biblioteka PHP dodająca obsługę uwierzytelniania wieloskładnikowego Duo Universal Prompt dla Zabbix.
Symfony Yaml Component		5.1.0	Dodaje obsługę eksportowania i importowania elementów konfiguracji Zabbix w formacie YAML.

Note:
Zabbix może również działać z wcześniejszymi wersjami Apache, MySQL i PostgreSQL.

Attention:
W przypadku czcionek innych niż domyślna DejaVu może być wymagana funkcja PHP imagerotate. Jeśli jej brakuje, czcionki te mogą być nieprawidłowo renderowane podczas wyświetlania wykresu. Funkcja ta jest dostępna tylko wtedy, gdy PHP zostało skompilowane z dołączonym GD, co nie ma miejsca w Debianie i innych dystrybucjach.

Biblioteki innych firm używane do tworzenia i debugowania kodu frontend Zabbix:

Biblioteka	Status wymagania	Minimalna wersja	Opis
Composer	Nie	2.4.1	Menedżer pakietów na poziomie aplikacji dla PHP, zapewniający standardowy format zarządzania zależnościami oprogramowania PHP i wymaganymi bibliotekami.
PHPUnit		8.5.29	Framework do testów jednostkowych PHP służący do testowania frontend Zabbix.
SASS		3.4.22	Język skryptowy preprocesora, który jest interpretowany i kompilowany do kaskadowych arkuszy stylów (CSS).

Przeglądarka internetowa po stronie klienta

Pliki cookie i JavaScript muszą być włączone.

Obsługiwane są najnowsze stabilne wersje Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari i Opera.

Warning:
Stosowana jest polityka same-origin dla IFrame, co oznacza, że Zabbix nie może być osadzony w ramkach w innej domenie.

 Jednak strony osadzone w ramce Zabbix będą miały dostęp do frontend Zabbix (przez JavaScript), jeśli strona osadzona w ramce i frontend Zabbix znajdują się w tej samej domenie. Strona taka jak `http://secure-zabbix.com/cms/page.html`, jeśli zostanie umieszczona na pulpitach w `http://secure-zabbix.com/zabbix/`, będzie miała pełny dostęp JS do Zabbix.

Serwer/proxy

Jeśli oprogramowanie/biblioteka zostały oznaczone jako wymagane, są bezwzględnie niezbędne. Opcjonalne elementy są potrzebne do obsługi określonych funkcji.

Wymaganie	Status obowiązkowy	Opis
libpcre2	Tak	Biblioteka PCRE2 jest wymagana do obsługi wyrażeń regularnych zgodnych z Perl (PCRE). Obsługiwana jest wersja PCRE2 v10.x.
libevent		Wymagana do komunikacji między procesami. Wersja 2.0.10 lub nowsza.
libevent-pthreads		Wymagana do komunikacji między procesami.
libpthread		Wymagana do obsługi mutexów oraz blokad odczytu i zapisu (może być częścią libc).
libresolv		Wymagana do rozpoznawania nazw DNS (może być częścią libc).
libiconv		Wymagana do konwersji kodowania/formatu tekstu (może być częścią libc). Obowiązkowa dla serwera Zabbix w systemie Linux.
libz		Wymagana do obsługi kompresji.
libm		Biblioteka matematyczna. Wymagana tylko przez serwer Zabbix.
libmysqlclient	Jedna z	Wymagana, jeśli używany jest MySQL.
libmariadb		Wymagana, jeśli używana jest MariaDB.
libpq5		Wymagana, jeśli używany jest PostgreSQL; wersja *libpq5* musi być taka sama lub nowsza niż wersja używanej bazy danych PostgreSQL.
libsqlite3		Wymagana, jeśli używany jest SQLite. Wymagana tylko przez proxy Zabbix.
libOpenIPMI	Nie	Wymagana do obsługi IPMI. Wymagana tylko przez serwer Zabbix.
libssh2 lub *libssh*		Wymagana do kontroli SSH. Wersja 1.8.0 lub nowsza (libssh2); 0.9.0 lub nowsza (libssh).
libcurl		Wymagana dla następujących funkcji: - monitorowanie sieci Web, monitorowanie VMware oraz pozycje typu agent HTTP (dla wszystkich: wersja 7.19.1 lub nowsza); - pozycje agenta Zabbix web.page.* (wersja 7.19.1 lub nowsza; zobacz także wymagania dotyczące agenta); - uwierzytelnianie SMTP (Basic: wersja 7.20.0 lub nowsza; OAuth: wersja 7.33 lub nowsza; zobacz także wymagania dotyczące frontend); - Elasticsearch (wersja 7.28.0 lub nowsza); - pozycje typu zapytanie telemetryczne (wersja 7.28.0 lub nowsza). Zalecana jest wersja 7.28.0 lub nowsza dla wszystkich funkcji. Aby używać zaktualizowanych funkcji cURL dla pozycji `web.page.*`, uruchom ponownie serwer/proxy Zabbix. W przypadku uwierzytelniania SMTP używaj pakietu `libcurl-full` w czasie działania.
libxml2		Wymagana do monitorowania VMware oraz wstępnego przetwarzania XML XPath.
net-snmp		Wymagana do obsługi SNMP. Wersja 5.3.0 lub nowsza. Obsługa protokołów silnego szyfrowania (AES192/AES192C, AES256/AES256C) jest dostępna od wersji biblioteki net-snmp 5.8; w systemach opartych na RHEL 8+ zaleca się używanie net-snmp 5.8.15 lub nowszej.
libunixodbc		Wymagana do monitorowania baz danych.
libgnutls lub *libopenssl*		Wymagana podczas korzystania z szyfrowania. Wersje minimalne: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1

Wymaganie	Status obowiązkowy	Opis
libldap		Wymagana do obsługi LDAP.
fping		Wymagana dla pozycji ping ICMP.
c-ares		Wymagana do asynchronicznego rozpoznawania nazw DNS, jeśli Zabbix został skonfigurowany z opcją `--with-ares`. W przeciwnym razie używana będzie biblioteka *libevent*. Wersja minimalna: 1.16.0. Funkcja buforowania zapytań DNS i przełączania awaryjnego (bufor zapytań jest domyślnie włączony) wymaga c-ares 1.26.0 lub nowszej. W systemach, w których c-ares jest dostarczana przez pakiet systemu operacyjnego, upewnij się, że zainstalowana biblioteka środowiska uruchomieniowego udostępnia funkcje resolvera wymagane do buforowania zapytań DNS i przełączania awaryjnego.
g++		Kompilator C++. Wymagany do kompilacji proxy Zabbix z opcją `--with-apm`.
gRPC		Wymagana do obsługi APM na proxy Zabbix. Wymagane biblioteki: *grpc++*, *grpc*, *gpr*. Narzędzie `grpc_cpp_plugin` jest również wymagane do kompilacji proxy Zabbix. Wymagana wersja jest dostępna w standardowych repozytoriach pakietów następujących systemów operacyjnych: - Ubuntu 23.04 lub nowszy; - Debian 12 (Bookworm) lub nowszy; - Alpine Linux 3.15 lub nowszy; - RHEL, Rocky Linux, CentOS Stream 9 lub nowszy (wyłącznie z repozytorium EPEL); - Fedora X lub nowsza. W starszych wersjach systemów operacyjnych *gRPC* należy zbudować ze źródeł.
protobuf		Wymagana do obsługi APM na proxy Zabbix. Kompilator `protoc` jest również wymagany do kompilacji proxy Zabbix.
Abseil		Wymagana do obsługi APM na proxy Zabbix. Wymagane biblioteki: *absl_synchronization*, *absl_base*. Zwykle instalowana razem z *gRPC*.

Agent

Requirement	Mandatory status	Description
libpcre2	Yes	Biblioteka PCRE2 jest wymagana do obsługi Perl Compatible Regular Expression (PCRE). Obsługiwana jest wersja PCRE2 v10.x. Wymagana do monitorowania logów. Wymagana również w systemie Windows.
libpthread		Wymagana do obsługi mutexów i blokad odczytu-zapisu (może stanowić część libc). Nie jest wymagana w systemie Windows.
libresolv		Wymagana do rozwiązywania DNS (może stanowić część libc). Nie jest wymagana w systemie Windows.
libiconv		Wymagana do konwersji kodowania/formatu tekstu do UTF-8 w pozycjach logów, zawartości plików, wyrażeniach regularnych plików oraz pozycjach regmatch (może stanowić część libc). Nie jest wymagana w systemie Windows.
libgnutls or *libopenssl*	No	Wymagana, jeśli używane jest szyfrowanie. Minimalne wersje: *libgnutls* - 3.1.18, *libopenssl* - 1.0.1 W systemie Microsoft Windows wymagany jest OpenSSL 1.1.1 lub nowszy.
libldap		Wymagana, jeśli używany jest LDAP. Nieobsługiwana w systemie Windows.
libcurl		Wymagana do rozszerzonej obsługi pozycji Zabbix agent web.page.*. Bez libcurl dostępna jest podstawowa funkcjonalność (np. web.page.get[http://example.com]). Z libcurl agent obsługuje dodatkowe funkcje, takie jak adresy URL HTTP z poświadczeniami (np. http://user:password@example.com) oraz adresy URL HTTPS. Wymagana jest wersja 7.19.1 lub nowsza (zalecana jest 7.28.0 lub nowsza). Aby korzystać z ulepszonych funkcji cURL, uruchom ponownie Zabbix agent.
libmodbus		Wymagana tylko wtedy, gdy używane jest monitorowanie Modbus. Wersja 3.0 lub nowsza.

Agent 2

Requirement	Mandatory status	Description
Go	Yes	Wymagane do zbudowania Zabbix agent 2 oraz jego wtyczek ze źródeł. Obsługiwany jest Go 1.25 lub nowszy. Instrukcje instalacji znajdują się na stronie go.dev. Biblioteki Go używane przez Zabbix agent 2 i jego wtyczki są wymienione w repozytorium Git Zabbix (biblioteki oznaczone jako `indirect` w repozytorium są zależnościami innych wymaganych bibliotek): - Zabbix agent 2 - Plugin support - PostgreSQL - MongoDB - MSSQL - Ember+ - NVIDIA GPU - Example plugin
libpcre2	Yes	Biblioteka PCRE2 jest wymagana do obsługi Perl Compatible Regular Expression (PCRE). Obsługiwana jest wersja PCRE2 v10.x. Wymagana do monitorowania logów. Wymagana również w systemie Windows.
libopenssl	No	Wymagana podczas korzystania z szyfrowania. Na platformach UNIX wymagana jest biblioteka OpenSSL 1.0.1 lub nowsza. Biblioteka OpenSSL musi mieć włączoną obsługę PSK. LibreSSL nie jest obsługiwany. W systemach Microsoft Windows wymagana jest biblioteka OpenSSL 1.1.1 lub nowsza.

Usługa web

Najnowsza stabilna wersja Google Chrome jest obsługiwana do generowania zaplanowanych raportów przy użyciu usługi web Zabbix.

Wymagana wersja Go do budowania usługi web jest taka sama jak ta używana dla Zabbix agent 2.

Java gateway

Jeśli Zabbix został pobrany z repozytorium źródłowego lub archiwum, niezbędne zależności są już zawarte w drzewie źródłowym.

Jeśli Zabbix został pobrany z pakietu dystrybucji, niezbędne zależności są już dostarczane przez system pakietów.

W obu powyższych przypadkach oprogramowanie jest gotowe do użycia i nie ma potrzeby pobierania dodatkowych plików.

Jeśli jednak chcesz dostarczyć własne wersje tych zależności (na przykład podczas przygotowywania pakietu dla określonej dystrybucji Linuksa), poniżej znajduje się lista wersji bibliotek, z którymi Java gateway jest znany ze współpracy. Zabbix może również działać z innymi wersjami tych bibliotek.

Poniższa tabela zawiera listę plików JAR, które są obecnie dołączone do Java gateway w oryginalnym kodzie:

Biblioteka	Status wymagania	Dołączona wersja	Komentarze
android-json	Tak	4.3r1	JSON (JavaScript Object Notation) to lekki format wymiany danych. Jest to zgodna z `org.json` implementacja systemu Android wyodrębniona z Android SDK.
logback-classic		1.5.25	
logback-core		1.5.25	
slf4j-api		2.0.16	

Java gateway można zbudować przy użyciu Oracle Java lub otwartoźródłowej implementacji OpenJDK (w wersji 8 lub nowszej). Pakiety dostarczane przez Zabbix są kompilowane przy użyciu OpenJDK. Poniższa tabela zawiera listę pakietów OpenJDK używanych do budowania pakietów Zabbix przez poszczególne dystrybucje:

Dystrybucja	Pakiet 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: 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)
Ubuntu 22.04	`default-jdk-headless` (amd64, arm64: 2:1.11-72build2)
Ubuntu 20.04	`default-jdk-headless` (amd64, arm64: 2:1.11-72)

Adapter mostka

Adapter mostka Zabbix nie ma specjalnych wymagań.

Domyślne porty

Poniższa lista zawiera porty, których komponenty Zabbix mogą używać w domyślnej konfiguracji:

Komponent Zabbix	Protokół i port	Połączenie inicjowane przez
agent Zabbix	TCP 10050	serwer Zabbix lub proxy Zabbix (tylko kontrole pasywne)
agent Zabbix 2	TCP 10050	serwer Zabbix lub proxy Zabbix (tylko kontrole pasywne)
serwer Zabbix	TCP 10051	Aktywne agenty Zabbix, aktywne proxy Zabbix, Zabbix sender, frontend Zabbix
proxy Zabbix	TCP 10051	Aktywne agenty Zabbix, Zabbix sender, serwer Zabbix (tylko pasywne proxy)
Java gateway Zabbix	TCP 10052	serwer Zabbix lub proxy

Komponent Zabbix	Protokół i port	Połączenie inicjowane przez
usługa internetowa Zabbix	TCP 10053	serwer Zabbix
serwer MCP Zabbix	TCP 8443	Klienci MCP (agenty AI)
frontend Zabbix	TCP 80	Przeglądarki internetowe, usługa internetowa Zabbix (raporty zaplanowane), klienci API
	TCP 443	
Pułapki SNMP (snmptrapd)	UDP 162	Urządzenia SNMP
Adapter mostu Zabbix	TCP 10054	serwer Zabbix

Warning:
Otwieraj tylko porty wymagane przez faktycznie uruchomione komponenty i ograniczaj każdą regułę zapory sieciowej do konkretnych hostów, które muszą się łączyć (na przykład zezwalaj na port 10050 tylko z serwera Zabbix lub proxy, a nie z dowolnego adresu). Nie udostępniaj portów Zabbix bezpośrednio w internecie.
 Przed otwarciem portów zabezpiecz instalację: włącz szyfrowanie między komponentami, ogranicz dozwolone hosty w plikach konfiguracyjnych komponentów i zapoznaj się z najlepszymi praktykami bezpieczeństwa.

Rozmiar bazy danych

Dane konfiguracyjne Zabbix wymagają stałej ilości miejsca na dysku i nie rosną znacząco.

Rozmiar bazy danych Zabbix zależy głównie od poniższych zmiennych, które określają ilość przechowywanych danych historycznych:

- Liczba przetwarzanych wartości na sekundę

Jest to średnia liczba nowych wartości, które serwer Zabbix otrzymuje co sekundę. Na przykład, jeśli mamy 3000 pozycji do monitorowania z częstotliwością odświeżania 60 sekund, liczba wartości na sekundę jest obliczana jako 3000/60 = **50**.

Oznacza to, że co sekundę do bazy danych Zabbix dodawanych jest 50 nowych wartości.

- Ustawienia housekeepera dla historii

Zabbix przechowuje wartości przez stały okres czasu, zwykle kilka tygodni lub miesięcy. Każda nowa wartość wymaga określonej ilości miejsca na dysku na dane i indeks.

Jeśli więc chcielibyśmy przechowywać historię przez 30 dni i otrzymujemy 50 wartości na sekundę, łączna liczba wartości wyniesie około (**30***24*3600)* **50** = 129.600.000, czyli około 130M wartości.

W zależności od używanego silnika bazy danych, typu otrzymywanych wartości (liczby zmiennoprzecinkowe, liczby całkowite, ciągi znaków, pliki dziennika itp.) miejsce na dysku potrzebne do przechowania pojedynczej wartości może wynosić od 40 bajtów do setek bajtów. Zwykle jest to około 90 bajtów na wartość dla pozycji liczbowych[2]. W naszym przypadku oznacza to, że 130M wartości będzie wymagać 130M * 90 bajtów = **10.9GB** miejsca na dysku.

Note:
Rozmiar wartości pozycji tekstowych/dziennika jest niemożliwy do dokładnego przewidzenia, ale można oczekiwać około 500 bajtów na wartość.

- Ustawienie housekeepera dla trendów

Zabbix przechowuje zestaw wartości max/min/avg/count z okresu 1 godziny dla każdej pozycji w tabeli **trends**. Dane są używane do tworzenia trendów i wykresów długookresowych. Okres jednej godziny nie może być dostosowany.

Baza danych Zabbix, w zależności od typu bazy danych, wymaga około 90 bajtów na każdy total. Załóżmy, że chcielibyśmy przechowywać dane trendów przez 5 lat. Wartości dla 3000 pozycji będą wymagać 3000*24*365* **90** = **2.2GB** rocznie, czyli **11GB** przez 5 lat.

- Ustawienia housekeepera dla zdarzeń

Każde zdarzenie Zabbix wymaga około 250 bajtów miejsca na dysku[1]. Trudno oszacować liczbę zdarzeń generowanych codziennie przez Zabbix. W najgorszym przypadku możemy założyć, że Zabbix generuje jedno zdarzenie na sekundę.

Dla każdego odzyskanego zdarzenia tworzony jest rekord event_recovery. Zwykle większość zdarzeń zostanie odzyskana, więc możemy założyć jeden rekord event_recovery na zdarzenie. Oznacza to dodatkowe 80 bajtów na zdarzenie.

Opcjonalnie zdarzenia mogą mieć tagi, a każdy rekord tagu wymaga około 100 bajtów miejsca na dysku[1]. Liczba tagów na zdarzenie (#tags) zależy od konfiguracji. Każdy z nich będzie więc wymagał dodatkowych #tags * 100 bajtów miejsca na dysku.

Oznacza to, że jeśli chcemy przechowywać zdarzenia przez 3 lata, będzie to wymagać 3*365*24*3600* (250+80+#tags*100) = **~30GB**+#tags*100B miejsca na dysku[2].

Note:
[1] Więcej w przypadku nazw zdarzeń, tagów i wartości spoza ASCII.
 [2] Przybliżenia rozmiaru są oparte na MySQL i mogą się różnić w przypadku innych baz danych.

Tabela zawiera wzory, które można wykorzystać do obliczenia miejsca na dysku wymaganego przez system 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.

Zatem całkowite wymagane miejsce na dysku można obliczyć jako:

Konfiguracja + Historia + Trendy + Zdarzenia

Miejsce na dysku NIE będzie używane natychmiast po instalacji Zabbix. Rozmiar bazy danych będzie rósł, a następnie w pewnym momencie przestanie rosnąć, co zależy od ustawień housekeepera.

Synchronizacja czasu

Bardzo ważne jest, aby na serwerze, na którym działa Zabbix, czas systemowy był precyzyjnie ustawiony. ntpd to najpopularniejszy demon synchronizujący czas hosta z czasem innych maszyn. Zdecydowanie zaleca się utrzymywanie zsynchronizowanego czasu systemowego na wszystkich systemach, na których działają komponenty Zabbix.

Co nowego i uwagi dotyczące aktualizacji: 8.0

Zabbix 8.0.0 jest kontynuacją Zabbix 7.4.0 i zawiera nowe funkcje oraz ulepszenia.

Na tej stronie również:

- Co nowego w szablonach, w tym nowe szablony i zmiany w istniejących szablonach.
- Uwagi dotyczące aktualizacji, w tym zmiany powodujące niezgodność.

Zobacz także: Zmiany w API.

Uwagi dotyczące aktualizacji

Te uwagi dotyczą aktualizacji z Zabbix 7.4.x do Zabbix 8.0.0.

Wszystkie uwagi są pogrupowane w:

- **Zmiany powodujące niezgodność** - zmiany, które mogą zakłócić działanie istniejących instalacji, oraz inne istotne informacje związane z procesem aktualizacji
- **Inne** - wszystkie pozostałe informacje opisujące zmiany w funkcjonalności Zabbix

Zobacz także:

- Procedura aktualizacji zawierająca wszystkie istotne informacje dotyczące aktualizacji z wersji wcześniejszych niż Zabbix 8.0.0;
- Aktualizacja klastra HA z instrukcjami dotyczącymi aktualizacji serwerów w klastrze **wysokiej dostępności** (HA).

Zmiany niekompatybilne wstecz

Inne

Co nowego i uwagi dotyczące aktualizacji: 8.0.x

Ta strona przedstawia wszystkie nowe funkcje zawarte w wydaniach pomniejszych głównej wersji Zabbix. Uwzględniono również zmiany w szablonach i uwagi dotyczące aktualizacji.

Zobacz także sekcję Co nowego dotyczącą głównej wersji.

Co nowego w Zabbix 8.0.1 Ta wersja nie została jeszcze wydana.

Zobacz uwagi dotyczące aktualizacji dla tej wersji.

Uwagi dotyczące aktualizacji

Urządzenie Zabbix

Przegląd Appliance Zabbix umożliwia natychmiastowe wdrożenie serwera Zabbix i frontend, zamiast konfigurowania ich ręcznie lub ponownego używania istniejącego serwera na potrzeby Zabbix.

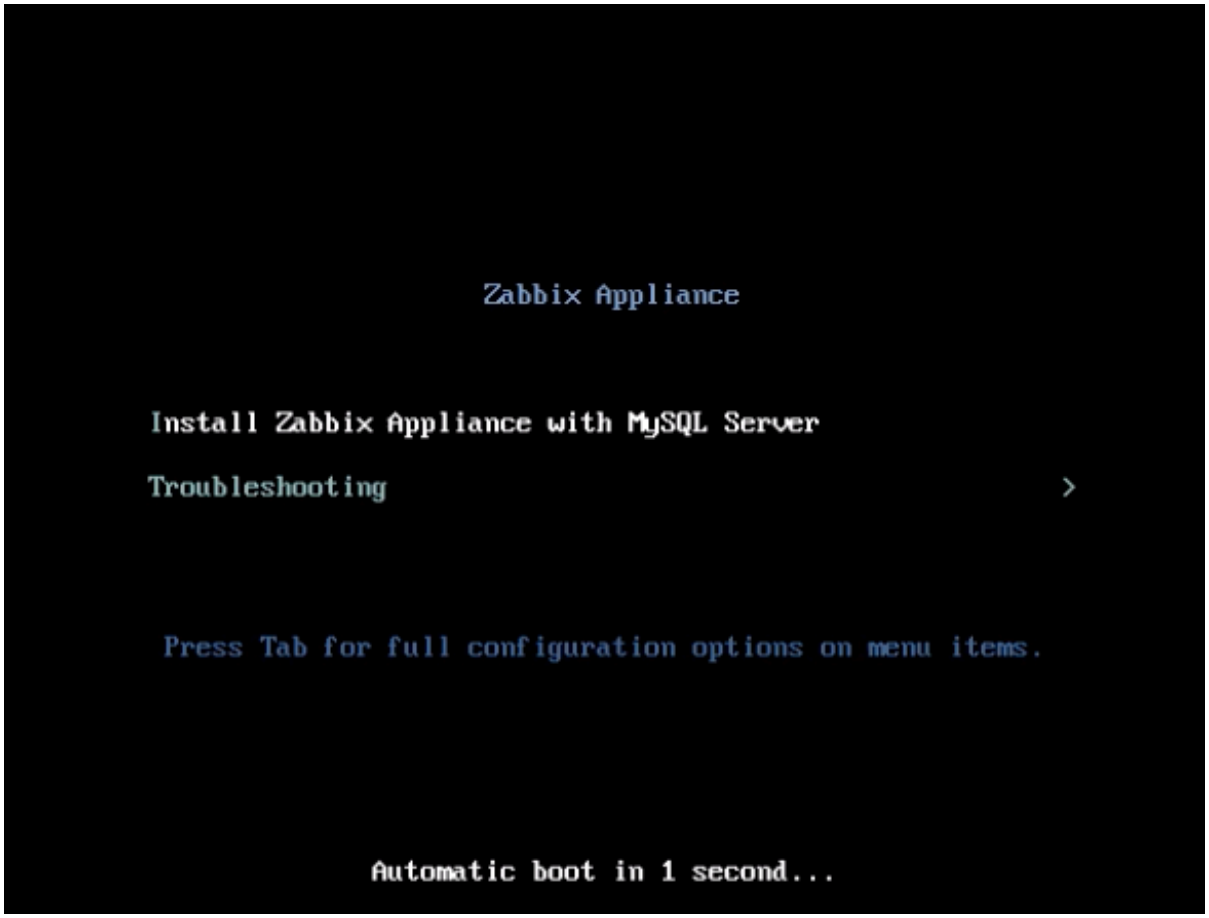
Appliance dla Zabbix 8.0 jest oparty na pochodnych systemu Red Hat Enterprise Linux 10, takich jak AlmaLinux lub Rocky Linux, i zawiera wstępnie skonfigurowany serwer Zabbix działający na MySQL oraz frontend działający na serwerze WWW NGINX.

> **Attention:**
> Ten appliance jest przeznaczony do oceny Zabbix. Nie zaleca się używania go w poważnych środowiskach produkcyjnych.

Obrazy appliance są dostępne do pobrania w następujących formatach:

- Instalacyjny CD/DVD (.iso)
- VMware (.vmx) - zobacz uwagi
- Open virtualization format (.ovf)
- Microsoft Hyper-V (.vhd/.vhdx) - zobacz uwagi
- KVM, Parallels, QEMU, pamięć USB, VirtualBox, Xen (.raw) - zobacz uwagi
- KVM, QEMU (.qcow2)

Menu rozruchowe instalacyjnego CD/DVD Zabbix:



Szybki start Wymagania wstępne

1. Upewnij się, że maszyna hosta ma wystarczające zasoby, aby spełnić wymagania systemowe maszyny wirtualnej:

- *RAM*: 4 GB
- *Miejsce na dysku*: dla maszyny wirtualnej należy przydzielić co najmniej 8 GB
- *CPU*: minimum 2 rdzenie

2. Jeśli oprogramowanie do wirtualizacji służące do uruchamiania obrazu appliance nie jest jeszcze zainstalowane, zainstaluj je (na przykład VirtualBox).
3. Pobierz appliance w formacie obsługiwanym przez używane oprogramowanie do wirtualizacji.
4. Sprawdź ustawienia sieciowe, aby upewnić się, że DHCP jest włączone na maszynie hosta.

Instalacja

1. Uruchom maszynę wirtualną appliance z pobranego obrazu. Instrukcje znajdziesz w dokumentacji oprogramowania do wirtualizacji, na przykład w dokumentacji VirtualBox.
2. Skonfiguruj ustawienia sieciowe maszyny wirtualnej, aby umożliwić dostęp z przeglądarki na hoście. Można to osiągnąć, włączając *tryb mostkowany*.
3. Zaloguj się do maszyny wirtualnej przy użyciu domyślnych danych uwierzytelniających.
4. Aby pobrać adres IP, uruchom na maszynie wirtualnej następujące polecenie:

```
ip addr show
```

5. Otwórz przeglądarkę na hoście i przejdź do adresu IP, który appliance otrzymał przez DHCP.
6. Zaloguj się do Zabbixa przy użyciu domyślnych danych uwierzytelniających i rozpocznij monitorowanie.

Konfiguracja Ta sekcja opisuje często wymagane domyślne ustawienia konfiguracji wraz z dostępnymi opcjami dostosowania.

Dane uwierzytelniające

System

- Nazwa użytkownika: `root`
- Hasło: `zabbix`

frontend Zabbix

- Nazwa użytkownika: `Admin`
- Hasło: `zabbix`

Po zalogowaniu możesz zmienić domyślne hasło w ustawieniach profilu użytkownika lub utworzyć nowych użytkowników i usunąć domyślnego użytkownika.

Baza danych

Hasła wszystkich użytkowników bazy danych są generowane losowo podczas procesu instalacji. W bazie danych zdefiniowano następujących użytkowników:

Root:

- Nazwa użytkownika: `root`
- Hasło: hasło jest przechowywane w pliku `/root/.my.cnf` Nie ma potrzeby wprowadzania hasła na koncie root.

serwer Zabbix:

- Nazwa użytkownika: `zabbix_srv`
- Hasło: hasło jest przechowywane w pliku konfiguracyjnym serwera

frontend Zabbix:

- Nazwa użytkownika: `zabbix_web`
- Hasło: hasło jest przechowywane w pliku konfiguracyjnym frontend

Aby zmienić hasło użytkownika bazy danych, zmodyfikuj je w MySQL oraz w odpowiednim pliku konfiguracyjnym.

Dostęp do frontend

Dostęp do frontend Zabbix można uzyskać pod adresem *http://<virtual machine's IP>*.

Domyślnie dostęp jest dozwolony z dowolnego miejsca. Aby go ograniczyć, zmodyfikuj plik `/etc/nginx/conf.d/zabbix.conf`. Po zapisaniu edytowanego pliku uruchom ponownie NGINX, logując się przez SSH jako użytkownik root i wykonując:

```
systemctl restart nginx
```

Statyczny adres IP

Domyślnie appliance używa DHCP do uzyskania adresu IP. Aby ustawić statyczny adres IP:

1. Zaloguj się jako użytkownik root.
2. Uruchom następujące polecenia, zastępując wartości własnym adresem IP:

```
nmcli con mod eth0 ipv4.addresses 192.168.1.10/24    # Adres IP appliance/prefiks CIDR
nmcli con mod eth0 ipv4.gateway 192.168.1.1     # Adres IP bramy
nmcli con mod eth0 ipv4.dns 8.8.8.8    # Adres IP serwera DNS
nmcli con mod eth0 ipv4.method manual
nmcli con up eth0
```

Alternatywnie możesz otworzyć plik `/etc/NetworkManager/system-connections/eth0.nmconnection` i dodać następujące wiersze:

```
[ipv4]
address1=192.168.1.10/24
dns=8.8.8.8;
gateway=192.168.1.1
method=manual
```

Po zapisaniu zmodyfikowanego pliku uruchom polecenia:

```
nmcli con reload
nmcli con up eth0
```

Aby sprawdzić, czy zmiany konfiguracji sieci zostały zastosowane, uruchom następujące polecenie:

```
nmcli dev show eth0
```

Konfiguracja zapory sieciowej

Do zarządzania ustawieniami zapory urządzenie appliance używa iptables ze wstępnie zdefiniowanymi regułami:

- Otwarty port SSH (22 TCP)
- Otwarte porty Zabbix agent (10050 TCP) i Zabbix trapper (10051 TCP)

- Otwarte porty HTTP (80 TCP) i HTTPS (443 TCP)
- Otwarty port pułapek SNMP (162 UDP)
- Otwarte połączenia wychodzące do portu NTP (123 UDP)
- Ograniczenie pakietów ICMP do 5 pakietów na sekundę
- Odrzucanie wszystkich pozostałych połączeń przychodzących

Aby otworzyć dodatkowe porty, zmodyfikuj plik `/etc/sysconfig/iptables` i przeładuj reguły zapory:

```
systemctl reload iptables
```

Repozytoria

Urządzenie Zabbix appliance używa pakietu *zabbix-release* z repozytorium Zabbix. Repozytoria są skonfigurowane w katalogu `/etc/yum.repos.d/*`.

Strefa czasowa

Domyślnie appliance używa UTC dla zegara systemowego. Aby zmienić strefę czasową, skopiuj odpowiedni plik z `/usr/share/zoneinfo` do `/etc/localtime`, na przykład:

```
cp /usr/share/zoneinfo/Europe/Riga /etc/localtime
```

> **Note:**
> Strefa czasowa frontendu Zabbix jest ustawiana oddzielnie i może zostać zmieniona w ustawieniach frontendu. Domyślna strefa czasowa dla frontendu Zabbix to Europe/Riga.

Lokalizacje plików

- Pliki konfiguracyjne serwera Zabbix i agenta znajdują się w `/etc/zabbix`.
- Pliki dziennika serwera Zabbix i agenta znajdują się w `/var/log/zabbix`.
- Plik konfiguracyjny frontendu Zabbix znajduje się w `/etc/zabbix/web`.
- Frontend Zabbix znajduje się w `/usr/share/zabbix`.
- Katalog domowy użytkownika *zabbix* to `/var/lib/zabbix`.

Usługi systemowe

Usługi systemd są dostępne. Aby wyświetlić listę usług Zabbix, uruchom następujące polecenie na maszynie wirtualnej:

```
systemctl list-units zabbix*
```

Uwagi dotyczące formatu obrazu VMware

Obrazy w formacie *vmdk* mogą być używane bezpośrednio w produktach VMware Player, Server i Workstation. Aby używać ich w ESX, ESXi i vSphere, należy je przekonwertować za pomocą narzędzia VMware vCenter Converter (pobieranie wymaga uwierzytelnienia). W przypadku korzystania z VMware vCenter Converter mogą wystąpić problemy z hybrydową kartą sieciową. W takim przypadku można spróbować określić kartę E1000 podczas procesu konwersji. Alternatywnie, po zakończeniu konwersji można usunąć istniejącą kartę i dodać kartę E1000.

Obraz HDD/flash (raw)

Aby uruchomić obraz, wykonaj:

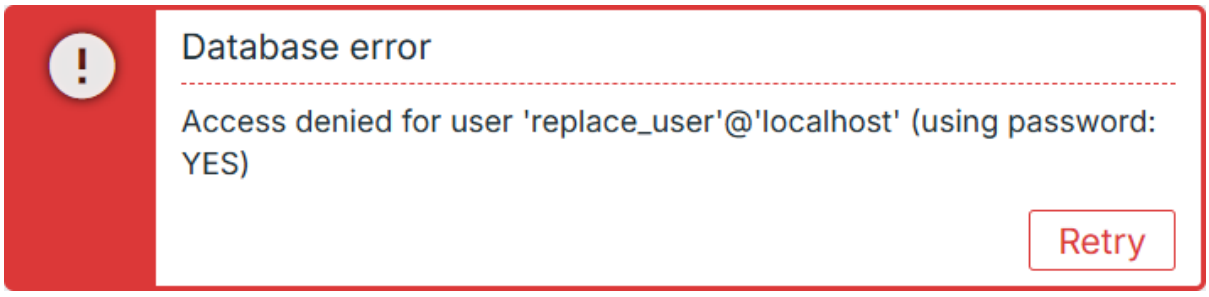
```
dd if=./zabbix_appliance_8.0.0.raw of=/dev/sdc bs=4k conv=fdatasync
```

Zastąp `/dev/sdc` ścieżką do urządzenia dysku Flash/HDD.

Hyper-V

Jeśli appliance nie uruchamia się w Hyper-V, spróbuj nacisnąć **Ctrl+Alt+F2**, aby przełączyć sesję TTY.

Rozwiązywanie problemów Jeśli podczas próby zalogowania się do frontend pojawi się komunikat o błędzie *Access denied for user 'replace_user'@'localhost' (using password: YES)*, może to oznaczać, że instalacja nadal trwa.



Jeśli błąd utrzymuje się po odczekaniu kilku minut lub zauważysz inne nieoczekiwane zachowanie, prawdopodobnie proces instalacji nie zakończył się pomyślnie. W takim przypadku zalecamy usunięcie bieżącego appliance i ponowne wdrożenie go zgodnie z tymi samymi instrukcjami instalacji. Ten krok zazwyczaj rozwiązuje problem.

Należy pamiętać, że ręczna naprawa uszkodzonej instalacji nie jest zalecana, ponieważ może prowadzić do dalszych komplikacji.

Procesy

Przegląd Zabbix składa się z kilku procesów, które współpracują ze sobą, aby monitorować systemy i wysyłać powiadomienia. Każdy proces pełni określoną rolę w infrastrukturze monitorowania.

Procesy podstawowe

- Serwer - centralny proces, który wykonuje odpytywanie i przechwytywanie danych, oblicza wyzwalacze oraz wysyła powiadomienia do użytkowników.
- Agent - zbiera dane z monitorowanych hostów.
- Agent 2 - agent nowej generacji z architekturą wtyczek i rozszerzonymi możliwościami.
- Proxy - zbiera dane monitorowania z monitorowanych urządzeń w imieniu serwera Zabbix i wysyła informacje do scentralizowanego serwera Zabbix w rozproszonym środowisku.
- Java gateway - umożliwia monitorowanie aplikacji Java.
- Serwer MCP - umożliwia agentom AI pracę z Zabbix przy użyciu protokołu Model Context Protocol.

Aby zapewnić łatwy dostęp do Zabbix z dowolnego miejsca i z dowolnej platformy, udostępniany jest interfejs internetowy. Interfejs jest częścią serwera Zabbix i zwykle (choć nie jest to konieczne) działa na tej samej maszynie fizycznej co serwer.

Wszystkie informacje konfiguracyjne, a także dane zebrane przez Zabbix, są przechowywane w bazie danych.

Nagłówek Nagłówek jest obecny we wszystkich wiadomościach żądań i odpowiedzi między komponentami Zabbix.
Jest wymagany do określenia długości wiadomości, tego, czy jest skompresowana, oraz tego, czy jest to duży pakiet.

Protokół komunikacji Zabbix ma limit rozmiaru pakietu 1 GB na połączenie. Limit 1 GB
jest stosowany zarówno do długości odebranych danych pakietu, jak i do długości danych nieskompresowanych.

Podczas wysyłania konfiguracji do Zabbix proxy limit rozmiaru pakietu jest zwiększany do 4 GB, aby umożliwić synchronizację dużych konfiguracji. Gdy długość danych przed kompresją przekracza 4 GB, serwer Zabbix automatycznie zaczyna używać formatu dużego pakietu (flaga 0x04), który zwiększa limit rozmiaru pakietu do 16 GB.

Należy pamiętać, że chociaż format dużego pakietu może być używany do wysyłania dowolnych danych, obecnie tylko synchronizator konfiguracji Zabbix proxy
może obsługiwać pakiety większe niż 1 GB.

Struktura nagłówka Nagłówek składa się z czterech pól. Wszystkie liczby w nagłówku są sformatowane w kolejności little-endian.

Field	Size	Size (large packet)	Description
<PROTOCOL>	4	4	"ZBXD" lub 5A 42 58 44
<FLAGS>	1	1	Flagi protokołu: 0x01 - protokół komunikacyjny Zabbix 0x02 - kompresja 0x04 - duży pakiet
<DATALEN>	4	8	Długość danych.
<RESERVED>	4	8	Gdy używana jest kompresja (flaga 0x02) - długość nieskompresowanych danych Gdy kompresja nie jest używana - 00 00 00 00

Przykłady nagłówka Poniżej znajdują się przykładowe fragmenty kodu pokazujące, jak dodać nagłówek protokołu Zabbix do danych, które chcesz wysłać, aby uzyskać pakiet, który należy wysłać do Zabbix, tak aby został poprawnie zinterpretowany. Te fragmenty kodu zakładają, że rozmiar danych nie przekracza 1 GB, więc nie jest używany format dużego pakietu.

Python

```
packet = b"ZBXD\1" + struct.pack("<II", len(data), 0) + data
```

lub

```
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;
```

lub

```
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;
```

lub

```
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"
```

Serwer

Omówienie Serwer Zabbix jest centralnym procesem oprogramowania Zabbix.

Serwer odpowiada za odpytywanie i przechwytywanie danych, oblicza wyzwalacze, wysyła powiadomienia do użytkowników. Jest centralnym komponentem, do którego agent i proxy Zabbix raportują dane o dostępności i integralności systemów. Serwer może również zdalnie sprawdzać usługi sieciowe (takie jak serwery WWW i serwery poczty) przy użyciu prostych testów usług.

Serwer jest centralnym repozytorium, w którym przechowywane są wszystkie dane konfiguracyjne, statystyczne i operacyjne, a także elementem Zabbix, który aktywnie powiadamia administratorów, gdy w którymkolwiek z monitorowanych systemów wystąpią problemy.

Działanie podstawowego serwera Zabbix jest podzielone na trzy odrębne komponenty; są to: serwer Zabbix, frontend i magazyn bazy danych.

Wszystkie informacje konfiguracyjne dla Zabbix są przechowywane w bazie danych, z którą współpracują zarówno serwer, jak i frontend. Na przykład, gdy tworzysz nową pozycja za pomocą frontend (lub API), jest ona dodawana do tabeli items w bazie danych. Następnie, mniej więcej raz na minutę, serwer Zabbix odpyta tabelę items o listę aktywnych pozycji, która zostanie następnie zapisana w pamięci podręcznej wewnątrz serwera Zabbix. Dlatego wprowadzone w frontend Zabbix zmiany mogą pojawić się w sekcji najnowszych danych dopiero po upływie nawet dwóch minut.

Zobacz także:

- Konfiguracja wysokiej dostępności dla serwerów Zabbix
- Parametry konfiguracji: serwer Zabbix

Uruchamianie serwera Jeśli zainstalowano jako pakiet

Serwer Zabbix działa jako proces demona. Serwer można uruchomić, wykonując:

```
systemctl start zabbix-server
```

To będzie działać w większości systemów GNU/Linux. W innych systemach może być konieczne uruchomienie:

```
/etc/init.d/zabbix-server start
```

Podobnie, aby zatrzymać, uruchomić ponownie lub wyświetlić stan, użyj następujących poleceń:

```
systemctl stop zabbix-server
systemctl restart zabbix-server
systemctl status zabbix-server
```

Uruchamianie ręczne

Jeśli powyższe nie działa, musisz uruchomić go ręcznie. Znajdź ścieżkę do pliku binarnego `zabbix_server` i wykonaj:

```
zabbix_server
```

Możesz używać następujących parametrów wiersza poleceń z serwerem Zabbix:

Parameter	Description
`-c --config <config-file>`	Ścieżka do pliku konfiguracyjnego.
`-f --foreground`	Uruchom serwer Zabbix na pierwszym planie.
`-R --runtime-control <option>`	Wykonaj funkcje administracyjne.
`-T --test-config`	Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.
`-h --help`	Wyświetl informacje pomocy i zakończ działanie.
`-V --version`	Wyświetl numer wersji.

Przykłady uruchamiania serwera Zabbix z parametrami wiersza poleceń:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf
zabbix_server --help
zabbix_server -V
```

Sterowanie w czasie działania

Opcje sterowania w czasie działania:

Opcja	Opis	Cel
`config_cache_reload`	Ponownie załaduj pamięć podręczną konfiguracji. Ignorowane, jeśli pamięć podręczna jest obecnie ładowana.	
`history_cache_clear[=<target>]`	Wyczyść pamięć podręczną historii dla pozycji określonej za pomocą jej identyfikatora. Dotyczy wszystkich wartości pozycji z wyjątkiem pierwszej i ostatniej.	**target** - identyfikator pozycji
`diaginfo[=<section>]`	Zbierz informacje diagnostyczne w pliku dziennika serwera.	`historycache` - statystyki pamięci podręcznej historii; `valuecache` - statystyki pamięci podręcznej wartości; `preprocessing` - statystyki menedżera przetwarzania wstępnego; `alerting` - statystyki menedżera alertów; `lld` - statystyki menedżera LLD; `locks` - lista muteksów (pusta w systemach *BSD*); `connector` - statystyki konektorów z największą kolejką.
`ha_status`	Zapisz stan klastra wysokiej dostępności (HA) w dzienniku.	
`ha_remove_node=target`	Usuń węzeł wysokiej dostępności (HA) określony za pomocą jego nazwy lub identyfikatora. Pamiętaj, że węzłów aktywnych/w trybie gotowości nie można usuwać.	**target** - nazwa lub identyfikator węzła (można go uzyskać, uruchamiając `ha_status`).
`ha_set_failover_delay=delay`	Ustaw opóźnienie przełączania awaryjnego wysokiej dostępności (HA). Obsługiwane są przyrostki czasu, np. 10s, 1m.	
`proxy_config_cache_reload[=target]`	Ponownie załaduj pamięć podręczną konfiguracji proxy.	**target** - rozdzielana przecinkami lista nazw proxy. Jeśli nie określono celu, ponownie załaduj konfigurację dla wszystkich proxy.
`secrets_reload`	Ponownie załaduj sekrety z Vault.	
`service_cache_reload`	Ponownie załaduj pamięć podręczną menedżera usług.	
`snmp_cache_reload`	Ponownie załaduj pamięć podręczną SNMP — wyczyść właściwości silnika SNMP (czas działania silnika, liczba uruchomień silnika, identyfikator silnika, dane uwierzytelniające) dla wszystkich hostów. Użyj tej opcji, aby wymusić globalne wyczyszczenie pamięci podręcznej podczas rozwiązywania problemów z SNMP.	

Opcja	Opis	Cel
`housekeeper_execute`	Uruchom procedurę porządkowania. Ignorowane, jeśli procedura porządkowania jest obecnie wykonywana.	
`trigger_housekeeper_execute`	Uruchom procedurę porządkowania wyzwalaczy. Ignorowane, jeśli procedura porządkowania wyzwalaczy jest obecnie wykonywana.	
`log_level_increase[=<target>]`	Zwiększ poziom rejestrowania; jeśli nie określono celu, zmiana dotyczy wszystkich procesów. Nieobsługiwane w systemach *BSD*.	**process type** - wszystkie procesy określonego typu (np. `poller`). Zobacz wszystkie typy procesów serwera. **process type,N** - typ i numer procesu (np. `poller,3`). **pid** - identyfikator procesu (1 do 65535). W przypadku większych wartości określ cel jako „process type,N”.
`log_level_decrease[=<target>]`	Zmniejsz poziom rejestrowania; jeśli nie określono celu, zmiana dotyczy wszystkich procesów. Nieobsługiwane w systemach *BSD*.	
`prof_enable[=<target>]`	Włącz profilowanie. Jeśli nie określono celu, dotyczy wszystkich procesów. Włączone profilowanie udostępnia szczegółowe informacje o wszystkich rwlock/muteksach według nazwy funkcji.	**process type** - wszystkie procesy określonego typu (np. `history syncer`) Obsługiwane typy procesów jako cele profilowania: `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** - typ i numer procesu (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). W przypadku większych wartości określ cel jako „process type,N”. **scope** - wartości `rwlock`, `mutex`, `processing` mogą być używane z typem i numerem procesu (np. `history syncer,1,processing`) lub ze wszystkimi procesami danego typu (np. `history syncer,rwlock`).
`prof_disable[=<target>]`	Wyłącz profilowanie. Jeśli nie określono celu, dotyczy wszystkich procesów.	**process type** - wszystkie procesy określonego typu (np. `history syncer`). Obsługiwane typy procesów jako cele profilowania: zobacz `prof_enable`. **process type,N** - typ i numer procesu (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). W przypadku większych wartości określ cel jako „process type,N”.

Gdy typ procesu jest określony jako cel sterowania w czasie działania, polecenie ma zastosowanie tylko do uruchomionych procesów tego typu. Jeśli żaden proces określonego typu nie jest obecnie uruchomiony, polecenie zwraca błąd:

```
zabbix_server -R log_level_increase='discovery manager'
Cannot redirect signal: "discovery manager" process does not exist
```

Przykład użycia sterowania w czasie działania do przeładowania pamięci podręcznej konfiguracji serwera:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R config_cache_reload
```

Przykłady użycia sterowania w czasie działania do przeładowania konfiguracji proxy:

```
### Przeładuj konfigurację wszystkich proxy:
zabbix_server -R proxy_config_cache_reload

### Przeładuj konfigurację Proxy1 i Proxy2:
zabbix_server -R proxy_config_cache_reload=Proxy1,Proxy2
```

Przykład użycia sterowania w czasie działania do wyczyszczenia pamięci podręcznej historii dla pozycji:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R history_cache_clear=42243
```

Przykłady użycia sterowania w czasie działania do zebrania informacji diagnostycznych:

```
### Zbierz wszystkie dostępne informacje diagnostyczne do pliku dziennika serwera:
zabbix_server -R diaginfo

### Zbierz statystyki pamięci podręcznej historii do pliku dziennika serwera:
zabbix_server -R diaginfo=historycache
```

Przykład użycia sterowania w czasie działania do przeładowania pamięci podręcznej SNMP:

```
zabbix_server -R snmp_cache_reload
```

Attention:
Gdy interfejs SNMPv3 jest aktualizowany za pośrednictwem interfejsu Zabbix UI, Zabbix w większości przypadków automatycznie przeładuje nowe poświadczenia SNMPv3 dla tego interfejsu; użyj `-R snmp_cache_reload` tylko wtedy, gdy odpytywanie nadal kończy się niepowodzeniem po zmianie poświadczeń (na przykład z powodu niespójności engineBoots/engineID lub urządzeń niezgodnych z RFC) albo gdy chcesz wymusić globalne wyczyszczenie pamięci podręcznej SNMP w celu rozwiązywania problemów.

Przykład użycia sterowania w czasie działania do uruchomienia housekeepera:

```
zabbix_server -c /usr/local/etc/zabbix_server.conf -R housekeeper_execute
```

Przykłady użycia sterowania w czasie działania do zmiany poziomu logowania:

```
### Zwiększ poziom logowania wszystkich procesów:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase

### Zwiększ poziom logowania drugiego procesu poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2

### Zwiększ poziom logowania procesu o PID 1234:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234

### Zmniejsz poziom logowania wszystkich procesów http poller:
zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_decrease="http poller"
```

Przykład ustawienia opóźnienia failover HA na minimalne 10 sekund:

```
zabbix_server -R ha_set_failover_delay=10s
```

Użytkownik procesu

Serwer Zabbix jest zaprojektowany do działania jako użytkownik inny niż root. Będzie działał jako dowolny użytkownik inny niż root, jako który został uruchomiony. Możesz więc uruchomić serwer jako dowolny użytkownik inny niż root bez żadnych problemów.

Jeśli spróbujesz uruchomić go jako 'root', przełączy się na zakodowanego na stałe użytkownika 'zabbix', który musi być obecny w Twoim systemie. Możesz uruchomić serwer jako 'root' tylko wtedy, gdy odpowiednio zmodyfikujesz parametr 'AllowRoot' w pliku konfiguracyjnym serwera.

Jeśli serwer Zabbix i agent są uruchomione na tym samym komputerze, zaleca się użycie innego użytkownika do uruchamiania serwera niż do uruchamiania agenta. W przeciwnym razie, jeśli oba działają jako ten sam użytkownik, agent może uzyskać dostęp do pliku konfiguracyjnego serwera, a każdy użytkownik na poziomie Admin w Zabbix może dość łatwo pobrać na przykład hasło do bazy danych.

Plik konfiguracyjny

Domyślna ścieżka do pliku konfiguracyjnego serwera zależy od systemu operacyjnego i metody instalacji Zabbix:

- W przypadku instalacji ze źródeł: `/usr/local/etc/zabbix_server.conf` lub ścieżka określona przez zmienne w czasie kompilacji `--sysconfdir` lub `--prefix`.
- W przypadku instalacji z pakietów i urządzenia: `/etc/zabbix/zabbix_server.conf`.

Szczegółowe informacje dotyczące konfigurowania `zabbix_server` znajdują się w opcjach pliku konfiguracyjnego.

Skrypty startowe

Skrypty są używane do automatycznego uruchamiania/zatrzymywania procesów Zabbix podczas uruchamiania/zamykania systemu. Skrypty znajdują się w katalogu misc/init.d.

Typy procesów i wątków serwera

- `agent poller` - asynchroniczny proces pollera dla pasywnych kontroli z wątkiem roboczym;
- `alert manager` - menedżer kolejki alertów;
- `alert syncer` - proces zapisujący alerty w bazie danych;
- `alerter` - proces wysyłający powiadomienia;
- `availability manager` - proces aktualizujący dostępność hostów;
- `browser poller` - poller dla kontroli pozycji przeglądarki;
- `configuration syncer` - proces zarządzający pamięci podręcznej danych konfiguracyjnych;
- `configuration syncer worker` - proces rozwiązujący i synchronizujący wartości makr użytkownika w nazwach pozycji;
- `connector manager` - proces zarządzający konektorami;
- `connector worker` - proces obsługujący żądania od menedżera konektorów;
- `discovery manager` - proces zarządzający wykrywaniem urządzeń;
- `discovery worker` - proces obsługujący zadania wykrywania od menedżera wykrywania;
- `escalator` - proces eskalacji działań;
- `ha manager` - proces zarządzający wysoką dostępnością;
- `history poller` - proces obsługujący kontrole obliczane wymagające połączenia z bazą danych;
- `history syncer` - proces zapisujący historię w bazie danych;
- `housekeeper` - proces usuwający nieaktualne dane (historię i trendy pozycji, sesje użytkowników, zdarzenia itp.), a także dane pozostałe po usuniętych obiektach;
- `http agent poller` - asynchroniczny proces pollera dla kontroli HTTP z wątkiem roboczym;
- `http poller` - poller monitorowania sieci WWW;
- `icmp pinger` - poller dla kontroli icmpping;
- `internal poller` - poller dla kontroli wewnętrznych;
- `ipmi manager` - menedżer pollerów IPMI;
- `ipmi poller` - poller dla kontroli IPMI;
- `java poller` - poller dla kontroli Java;
- `lld manager` - proces zarządzający zadaniami wykrywania niskopoziomowego;
- `lld worker` - proces roboczy zadań wykrywania niskopoziomowego;
- `odbc poller` - poller dla kontroli ODBC;
- `poller` - zwykły poller dla pasywnych kontroli;
- `preprocessing manager` - menedżer zadań wstępnego przetwarzania z wątkami roboczymi wstępnego przetwarzania;
- `preprocessing worker` - wątek wstępnego przetwarzania danych;
- `proxy poller` - poller dla pasywnych proxy;
- `proxy group manager` - menedżer równoważenia obciążenia i wysokiej dostępności proxy;
- `report manager` - menedżer zaplanowanych zadań generowania raportów;
- `report writer` - proces generujący zaplanowane raporty;
- `self-monitoring` - proces zbierający wewnętrzne statystyki serwera;
- `service manager` - proces zarządzający usługami poprzez odbieranie informacji o problemach, tagach problemów i odzyskaniu po problemie z history syncer, task manager i alert manager;
- `snmp poller` - asynchroniczny proces pollera dla kontroli SNMP z wątkiem roboczym (tylko pozycje `walk[OID]` i `get[OID]`);
- `snmp trapper` - trapper dla trapów SNMP;
- `supervisor` - koordynuje uruchamianie i zatrzymywanie procesów oraz wątków serwera, a także monitoruje ich gotowość;
- `task manager` - proces zdalnego wykonywania zadań żądanych przez inne komponenty (np. zamykanie problemu, potwierdzanie problemu, natychmiastowe sprawdzanie wartości pozycji, funkcjonalność zdalnych poleceń);
- `telemetry query poller` - poller dla kontroli pozycji zapytań telemetrycznych;
- `timer` - zegar przetwarzający okresy utrzymania;
- `trapper` - trapper dla aktywnych kontroli, trapów i komunikacji z proxy;
- `trigger housekeeper` - proces usuwający problemy i zdarzenia wygenerowane przez wyzwalacze, które zostały później usunięte;
- `unreachable poller` - poller dla niedostępnych urządzeń;
- `vmware collector` - kolektor danych VMware odpowiedzialny za zbieranie danych z usług VMware.

Plik dziennika serwera może służyć do obserwowania tych typów procesów.

Plik dziennika serwera jest tworzony z uprawnieniami odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo plik jest dostępny do odczytu dla grupy właściciela. Wszystkie pozostałe uprawnienia są odrzucane.

Różne typy procesów serwera Zabbix można monitorować za pomocą wewnętrznej pozycji `zabbix[process,<type>,<mode>,<state>]`.

Statystyki transakcji synchronizatora historii

Tytuł procesu synchronizatora historii wyświetla szczegółowe statystyki dotyczące transakcji synchronizatora historii:

```
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
```

W sekcji "A+B triggers":

- A - wyzwalacze przetworzone na podstawie wartości historii;
- B - wyzwalacze przetworzone na podstawie timerów.

Czasy w `processed...in N (<timings>) sec` to:

- Czas spędzony na zapisywaniu wartości pozycji do bazy danych;
- Czas spędzony na aktualizowaniu danych pozycji (stan, błędy, inwentarz hosta itp.);
- Czas spędzony na zapisywaniu trendów do bazy danych;
- Czas spędzony na obliczaniu wyzwalaczy;
- Czas spędzony na przetwarzaniu zdarzeń i działań.

Procedura housekeeping

Proces `housekeeper` okresowo usuwa nieaktualne dane (historię i trendy pozycji, sesje użytkowników, zdarzenia itp.), a także dane pozostawione przez usunięte obiekty. Działa on w cyklach, a częstotliwość oraz limit usuwania na cykl są określane przez `HousekeepingFrequency` i `MaxHousekeeperDelete`. Wszelkie dane nieusunięte w jednym cyklu są przenoszone do następnego. Automatyczny housekeeping można włączyć i skonfigurować w *Administration* > *Housekeeping*.

Gdy proces `housekeeper` usuwa dane pozostawione przez usunięte obiekty, korzysta z zadań z tabeli `housekeeper`, która jest wypełniana za każdym razem, gdy obiekt zostanie usunięty. Na przykład po usunięciu hosta Zabbix usuwa jego pozycje, ale nie ich historię, trendy ani problemy. Zamiast tego wyzwalacze bazy danych wypełniają tabelę `housekeeper` zadaniami zawierającymi następujące pola:

- `housekeeperid` - ID zadania
- `object` - typ obiektu (0 - pozycja; 1 - wyzwalacz; 2 - usługa; 3 - wykryty host; 4 - wykryta usługa)
- `objectid` - ID obiektu (pomaga procesowi housekeeper znaleźć dane powiązane z obiektem)

Na przykład usunięcie hosta z dwiema pozycjami i jednym wyzwalaczem wypełnia tabelę `housekeeper` w następujący sposób:

```
+---------------+--------+----------+
| housekeeperid | object | objectid |
+---------------+--------+----------+
|             1 |      1 |    28724 |
|             2 |      0 |    59396 |
|             3 |      0 |    59397 |
+---------------+--------+----------+
```

Wyzwalacze bazy danych wypełniają tabelę `housekeeper` zadaniami bez sprawdzania danych powiązanych z obiektem; to sprawdzenie jest wykonywane przez proces `housekeeper` podczas usuwania danych:

- Dla usuniętych pozycji (w tym reguł LLD) - usuwa ich dane ze wszystkich tabel historii i trendów (`history`, `history_str`, `history_log`, `history_uint`, `history_text`, `history_bin`, `history_json`, `trends`, `trends_uint`). Usuwa również ich zdarzenia wewnętrzne z tabeli `problem`.
- Dla usuniętych wyzwalaczy - usuwa ich zdarzenia z tabel związanych ze zdarzeniami (`problem`, `event_symptom`, `events`; sama baza danych usuwa również zdarzenia z tabeli `event_recovery`). Powiadamia również proces `service manager` o usuniętych zdarzeniach.

Note:
Oddzielny proces `trigger housekeeper` obsługuje węższe zadanie — usuwanie problemów i zdarzeń, które nie mają znanego źródłowego wyzwalacza. Częstotliwość jego działania jest kontrolowana przez `ProblemHousekeepingFrequency`.
Do czasu uruchomienia procedury housekeeping dla wyzwalaczy problemy utworzone przez wyzwalacze, które zostały później usunięte, mogą nadal generować problemy usług i przypisywać je do usług. Jeśli Twoja konfiguracja obejmuje wiele reguł obliczania statusu usług opartych na często wykrywanych/nieodkrywanych wyzwalaczach, rozważ zwiększenie częstotliwości procedury housekeeping przez dostosowanie parametru konfiguracji serwera `ProblemHousekeepingFrequency`.

- Dla usuniętych usług - usuwa ich problemy z tabeli `problem`; ich zdarzenia z tabeli `events` są usuwane dopiero po upływie skonfigurowanego okresu przechowywania danych.
- Dla usuniętych wykrytych hostów/usług (network discovery) - usuwa ich zdarzenia z tabeli `events`.

Proces `housekeeper` usuwa tylko te zdarzenia, które nie są powiązane z problemami. Na przykład przestarzałe zdarzenie problemu/odzyskania nie zostanie usunięte, jeśli jest powiązane z otwartym problemem. Gdy `housekeeper` usuwa przestarzałe encje, najpierw usuwa problemy, a następnie zdarzenia.

Note:
Tabele używające trybu `partition` (partycjonowane tabele TimescaleDB) są pomijane; przetwarzane są tylko tabele używające trybu `regular`.

Obsługiwane platformy Ze względu na wymagania bezpieczeństwa i krytyczny charakter działania serwera, UNIX jest jedynym systemem operacyjnym, który może konsekwentnie zapewnić niezbędną wydajność, tolerancję błędów i odporność. Zabbix działa na wiodących wersjach rynkowych.

Serwer Zabbix jest testowany na następujących platformach:

- Linux
- Solaris
- AIX
- HP-UX
- Mac OS X
- FreeBSD
- OpenBSD
- NetBSD
- SCO Open Server

Note:
Zabbix może również działać na innych systemach operacyjnych podobnych do Unix.

Ustawienia regionalne Należy pamiętać, że serwer wymaga ustawień regionalnych UTF-8, aby niektóre elementy tekstowe mogły być interpretowane poprawnie. Większość nowoczesnych systemów typu Unix ma domyślnie ustawione ustawienia regionalne UTF-8, jednak w niektórych systemach może być konieczne ich jawne skonfigurowanie.

Konfiguracja wysokiej dostępności dla serwerów Zabbix

Omówienie

Wysoka dostępność (HA) jest zazwyczaj wymagana w infrastrukturach krytycznych, które mogą sobie pozwolić na praktycznie zerowy czas przestoju. Dlatego dla każdej usługi, która może ulec awarii, musi istnieć opcja przełączenia awaryjnego, aby przejąć działanie w przypadku awarii bieżącej usługi.

Zabbix oferuje **natywne** rozwiązanie wysokiej dostępności, które jest łatwe w konfiguracji i nie wymaga wcześniejszego doświadczenia z HA. Natywna HA Zabbixa może być przydatna jako dodatkowa warstwa ochrony przed awariami oprogramowania/sprzętu serwera Zabbix lub w celu skrócenia przestojów związanych z konserwacją.

W trybie wysokiej dostępności Zabbixa wiele serwerów Zabbix działa jako węzły w klastrze. Gdy jeden serwer Zabbix w klastrze jest aktywny, pozostałe pozostają w trybie gotowości, przygotowane do przejęcia działania w razie potrzeby.



Przejście na HA Zabbixa nie jest wiążące. W dowolnym momencie można wrócić do pracy w trybie samodzielnym.

Zobacz także: Szczegóły implementacji

Włączanie wysokiej dostępności

Uruchamianie serwera Zabbix jako węzła klastra

Do uruchomienia serwera Zabbix jako węzła klastra wymagane są dwa parametry w konfiguracji serwera:

- Parametr `HANodeName` musi być określony dla każdego serwera Zabbix, który będzie węzłem klastra HA.

Jest to unikatowy identyfikator węzła (np. `zabbix-node-01`), za pomocą którego serwer będzie identyfikowany w konfiguracjach agentów i proxy. Jeśli nie określisz parametru `HANodeName`, serwer zostanie uruchomiony w trybie autonomicznym.

- Parametr `NodeAddress` musi być określony dla każdego węzła.

Parametr `NodeAddress` (`address:port`) będzie używany przez frontend Zabbix do łączenia się z aktywnym węzłem serwera. `NodeAddress` musi odpowiadać adresowi IP lub nazwie FQDN odpowiedniego serwera Zabbix.

Po wprowadzeniu zmian w plikach konfiguracyjnych uruchom ponownie wszystkie serwery Zabbix. Zostaną one teraz uruchomione jako węzły klastra. Nowy status serwerów można sprawdzić w sekcji *Raporty* > *Informacje o systemie*, a także za pomocą polecenia:

```
zabbix_server -R ha_status
```

To polecenie runtime zapisze bieżący status klastra HA w logu serwera Zabbix (oraz na standardowym wyjściu):

```
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
```

Przygotowanie frontendu

Upewnij się, że `address:port` serwera Zabbix nie jest zdefiniowany w pliku konfiguracji frontendu Zabbix (`zabbix.conf.php`).

```
// Uncomment and set to desired values to override Zabbix hostname/IP and port.
// $ZBX_SERVER                  = '';
// $ZBX_SERVER_PORT             = '';
```

Frontend Zabbix automatycznie wykryje aktywny węzeł, odczytując ustawienia z tabeli węzłów w bazie danych Zabbix. Adres węzła aktywnego węzła zostanie użyty jako adres serwera Zabbix.

Konfiguracja proxy

Węzły klastra HA (serwery) muszą być wymienione w konfiguracji pasywnego lub aktywnego proxy Zabbix.

W przypadku pasywnego proxy nazwy węzłów muszą być wymienione w parametrze Server parameter proxy, oddzielone **przecinkiem**.

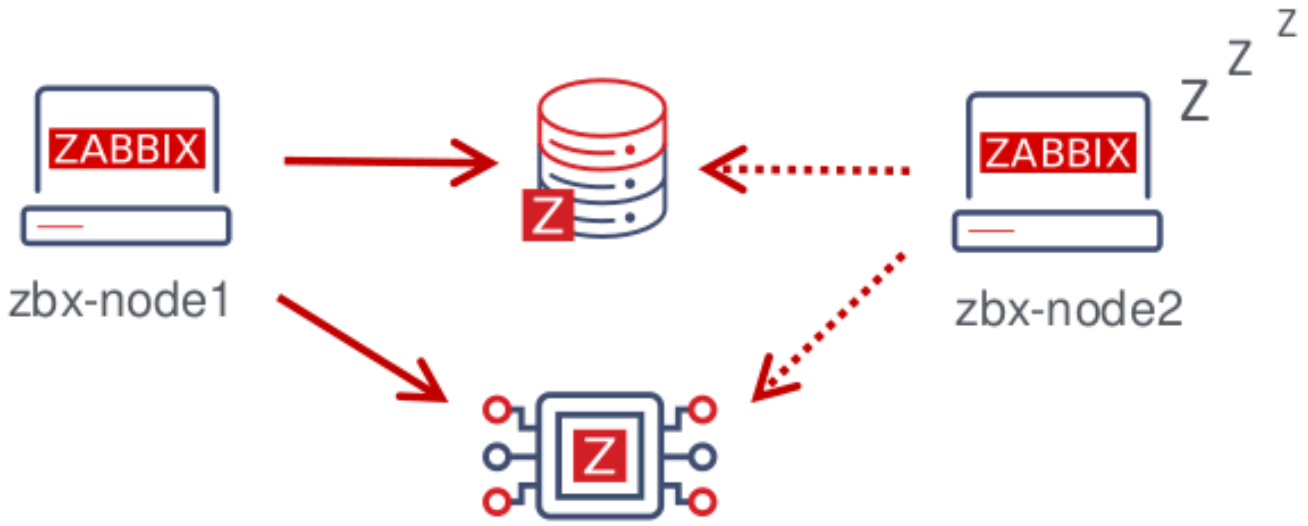
```
Server=zabbix-node-01,zabbix-node-02
```

W przypadku aktywnego proxy nazwy węzłów muszą być wymienione w parametrze Server parameter proxy, oddzielone **średnikiem**.

```
Server=zabbix-node-01;zabbix-node-02
```

Konfiguracja agenta

Węzły klastra HA (serwery) muszą być wymienione w konfiguracji Zabbix agent lub Zabbix agent 2.



Aby włączyć pasywne sprawdzenia, nazwy węzłów muszą być wymienione w parametrze Server parameter, oddzielone **przecinkiem**.

```
Server=zabbix-node-01,zabbix-node-02
```

Aby włączyć aktywne sprawdzenia, nazwy węzłów muszą być wymienione w parametrze ServerActive parameter. Należy pamiętać, że w przypadku aktywnych sprawdzeń węzły muszą być oddzielone przecinkiem od innych serwerów, natomiast same węzły muszą być oddzielone **średnikiem**, np.:

```
ServerActive=zabbix-node-01;zabbix-node-02
```

Przełączenie awaryjne na węzeł rezerwowy

Zabbix automatycznie przełączy się na inny węzeł, jeśli aktywny węzeł przestanie działać. Aby przełączenie awaryjne mogło nastąpić, musi istnieć co najmniej jeden węzeł o statusie rezerwowym.

Jak szybko nastąpi przełączenie awaryjne? Wszystkie węzły aktualizują czas ostatniego dostępu (oraz status, jeśli uległ zmianie) co 5 sekund. Zatem:

- Jeśli aktywny węzeł zostanie zamknięty i zdoła zgłosić swój status jako "stopped", inny węzeł przejmie działanie w ciągu **5 sekund**.
- Jeśli aktywny węzeł zostanie zamknięty / stanie się niedostępny bez możliwości aktualizacji swojego statusu, węzły rezerwowe będą czekać **opóźnienie przełączenia awaryjnego** + 5 sekund przed przejęciem działania.

Opóźnienie przełączenia awaryjnego jest konfigurowalne, a obsługiwany zakres wynosi od 10 sekund do 15 minut (domyślnie jedna minuta). Aby zmienić opóźnienie przełączenia awaryjnego, możesz uruchomić:

```
zabbix_server -R ha_set_failover_delay=5m
```

Zarządzanie klastrem HA

Bieżący stan klastra HA można zarządzać za pomocą dedykowanych opcji sterowania w czasie działania:

- `ha_status` - rejestrowanie stanu klastra HA w dzienniku serwera Zabbix (oraz na standardowym wyjściu);
- `ha_remove_node=target` - usunięcie węzła HA zidentyfikowanego za pomocą jego `<target>` - nazwy lub identyfikatora węzła (nazwę/identyfikator można uzyskać z danych wyjściowych polecenia `ha_status`), np.:

```
zabbix_server -R ha_remove_node=zabbix-node-02
```

Należy pamiętać, że aktywnych węzłów ani węzłów w trybie standby nie można usuwać.

- `ha_set_failover_delay=delay` - ustawienie opóźnienia przełączania awaryjnego HA (od 10 sekund do 15 minut; obsługiwane są przyrostki czasu (np. `30s`, `1m`)).

Stan węzłów można monitorować:

- W sekcji *Raporty* > *Informacje o systemie*.
- W widżecie pulpitu nawigacyjnego *Informacje o systemie*.
- Za pomocą opcji sterowania w czasie działania `ha_status` serwera (zobacz powyżej).

Wewnętrzna pozycja `zabbix[cluster,discovery,nodes]` może służyć do wykrywania węzłów, ponieważ zwraca dane JSON zawierające informacje o węzłach wysokiej dostępności.

Wyłączanie klastra HA

Aby wyłączyć klaster wysokiej dostępności:

1. Utwórz kopie zapasowe plików konfiguracyjnych.
2. Zatrzymaj węzły rezerwowe.
3. Usuń parametr HANodeName z aktywnego serwera głównego.
4. Uruchom ponownie serwer główny (zostanie uruchomiony w trybie autonomicznym).

Aktualizacja klastra HA

Aby wykonać aktualizację do głównej wersji dla węzłów HA:

1. Zatrzymaj wszystkie węzły.
2. Utwórz pełną kopię zapasową bazy danych.
3. Jeśli baza danych używa replikacji, upewnij się, że wszystkie węzły są zsynchronizowane i nie występują żadne problemy. Nie wykonuj aktualizacji, jeśli replikacja jest uszkodzona.
4. Wybierz jeden węzeł, który wykona aktualizację bazy danych, zmień jego konfigurację na tryb standalone, komentując `HANodeName`, i zaktualizuj go.
5. Upewnij się, że aktualizacja bazy danych została w pełni zakończona (*System information* powinno wyświetlać, że serwer Zabbix działa).
6. Uruchom ponownie węzeł w trybie HA.
7. Zaktualizuj i uruchom pozostałe węzły (nie ma potrzeby przełączania ich do trybu standalone, ponieważ baza danych jest już w tym momencie zaktualizowana).

W przypadku aktualizacji do wersji minor wystarczy zaktualizować pierwszy węzeł, upewnić się, że został zaktualizowany i działa, a następnie rozpocząć aktualizację następnego węzła.

Szczegóły implementacji

Klaster wysokiej dostępności (HA) jest rozwiązaniem opcjonalnym i jest obsługiwany dla serwera Zabbix. Natywne rozwiązanie HA zostało zaprojektowane tak, aby było proste w użyciu, działało między lokalizacjami i nie miało szczególnych wymagań dotyczących baz danych rozpoznawanych przez Zabbix. Użytkownicy mogą korzystać z natywnego rozwiązania HA Zabbix albo z rozwiązania HA innej firmy, w zależności od tego, co najlepiej odpowiada wymaganiom wysokiej dostępności w ich środowisku.

Rozwiązanie składa się z wielu instancji lub węzłów `zabbix_server`. Każdy węzeł:

- Jest konfigurowany oddzielnie.
- Korzysta z tej samej bazy danych.
- Może mieć kilka trybów: active, standby, unavailable, stopped.

Tylko jeden węzeł może być aktywny (working) w danym momencie. Węzeł standby uruchamia tylko jeden proces - menedżer HA. Węzeł standby nie wykonuje zbierania danych, przetwarzania ani innych standardowych działań serwera; nie nasłuchuje na portach; ma minimalną liczbę połączeń z bazą danych.

Zarówno węzły active, jak i standby aktualizują swój czas ostatniego dostępu co 5 sekund. Każdy węzeł standby monitoruje czas ostatniego dostępu węzła active. Jeśli czas ostatniego dostępu węzła active przekroczy liczbę sekund 'failover delay', węzeł standby sam przełącza się na węzeł active i przypisuje status `unavailable` wcześniej aktywnemu węzłowi.

Węzeł active monitoruje własną łączność z bazą danych - jeśli zostanie utracona na dłużej niż `failover delay`−5 sekund, musi zatrzymać całe przetwarzanie i przełączyć się w tryb standby. Węzeł active monitoruje również status węzłów standby - jeśli czas ostatniego dostępu węzła standby przekroczy `failover delay` sekund, węzłowi standby zostaje przypisany status `unavailable`.

Węzły zostały zaprojektowane tak, aby były zgodne między wersjami pomocniczymi Zabbix.

Parametry konfiguracji: serwer Zabbix

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania serwera Zabbix.

Wartości parametrów należy określić w pliku konfiguracji serwera Zabbix (`zabbix_server.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona jest podzielona na:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
AlertScriptsPath	Określa lokalizację skryptów alertów.
AllowRoot	Zezwala serwerowi na uruchamianie jako `root`.
AllowSoftwareUpdateCheck	Zezwala na odbieranie informacji o aktualizacjach oprogramowania z `zabbix.com`.
AllowUnsupportedDBVersions	Zezwala serwerowi na pracę z nieobsługiwanymi wersjami bazy danych.
BridgeAdapterURL	Ustawia adres URL HTTP(S) adaptera mostu Zabbix.
CacheSize	Ustawia rozmiar pamięci podręcznej konfiguracji (w bajtach).
CacheUpdateFrequency	Ustawia częstotliwość aktualizacji pamięci podręcznej konfiguracji (w sekundach).
DBHost	Określa host bazy danych (lub katalog gniazda).
DBName	Określa nazwę bazy danych. **Wymagane**.
DBPassword	Określa hasło do bazy danych.
DBPort	Określa port hosta bazy danych.
DBSchema	Określa nazwę schematu bazy danych. Używane w przypadku PostgreSQL.
DBSocket	Określa ścieżkę do pliku gniazda MySQL.
DBUser	Określa użytkownika bazy danych.
DBTLSConnect	Określa wartość wymuszającą połączenie TLS z bazą danych.
DBTLSCAFile	Określa pełną nazwę ścieżki pliku zawierającego certyfikaty najwyższego poziomu CA do weryfikacji certyfikatu bazy danych.
DBTLSCertFile	Określa pełną nazwę ścieżki pliku zawierającego certyfikat serwera Zabbix używany do uwierzytelniania w bazie danych.
DBTLSKeyFile	Określa pełną nazwę ścieżki pliku zawierającego klucz prywatny używany do uwierzytelniania w bazie danych.
DBTLSCipher	Określa listę szyfrów szyfrowania dozwolonych przez serwer Zabbix dla protokołów TLS do wersji TLS 1.2 włącznie. Obsługiwane tylko w przypadku MySQL.
DBTLSCipher13	Określa listę zestawów szyfrów dozwolonych przez serwer Zabbix dla protokołu TLS 1.3. Obsługiwane tylko w przypadku MySQL, począwszy od wersji 8.0.16.
DebugLevel	Ustawia poziom debugowania.
EnableGlobalScripts	Włącza skrypty globalne na serwerze Zabbix.
EnableMobileDevices	Włącza obsługę urządzeń mobilnych na serwerze Zabbix.
ExportDir	Określa katalog w celu włączenia eksportu zdarzeń, danych historycznych i trendów w czasie rzeczywistym.
ExportFileSize	Ustawia maksymalny rozmiar pliku eksportu (w bajtach).
ExportType	Określa typy encji eksportowanych w czasie rzeczywistym (zdarzenia, dane historyczne, trendy).
ExternalScripts	Określa lokalizację skryptów zewnętrznych.
Fping6Location	Określa lokalizację fping6.
FpingLocation	Określa lokalizację fping.
FrontendAllowedIP	Określa adresy IP lub zakresy CIDR, z których frontend może nawiązywać połączenia.
HANodeName	Określa nazwę węzła w klastrze wysokiej dostępności.
HistoryCacheSize	Ustawia rozmiar pamięci podręcznej danych historycznych.
HistoryIndexCacheSize	Ustawia rozmiar pamięci podręcznej indeksów danych historycznych.
HistoryProvider	Ustawia dostawcę przechowywania danych historycznych (Elasticsearch, ClickHouse lub SQL).
HistoryStorageDateIndex	Ten parametr jest **przestarzały** (zamiast niego użyj HistoryProvider). Włącza przechowywanie wartości historycznych w różnych indeksach na podstawie daty.
HistoryStorageURL	Ten parametr jest **przestarzały** (zamiast niego użyj HistoryProvider). Określa adres URL magazynu danych historycznych.
HistoryStorageTypes	Ten parametr jest **przestarzały** (zamiast niego użyj HistoryProvider). Określa typy wartości wysyłanych do magazynu danych historycznych.
HousekeepingFrequency	Ustawia częstotliwość procedury porządkowania (w godzinach).
Include	Określa katalog lub pojedyncze pliki do uwzględnienia w pliku konfiguracji.
JavaGateway	Określa adres IP (lub nazwę hosta) Java gateway Zabbix.
JavaGatewayPort	Określa port Java gateway Zabbix.
ListenBacklog	Ustawia maksymalną liczbę oczekujących połączeń w kolejce TCP.
ListenIP	Określa adres IP, na którym nasłuchuje trapper.
ListenPort	Określa port, na którym nasłuchuje trapper.
LoadModule	Określa moduł ładowany podczas uruchamiania serwera.
LoadModulePath	Określa pełną ścieżkę do lokalizacji modułów serwera.
LogFile	Określa plik dziennika serwera.
LogFileSize	Ustawia maksymalny rozmiar pliku dziennika.
LogSlowQueries	Ustawia maksymalny czas trwania zapytania do bazy danych, po którego przekroczeniu zapytanie zostanie zapisane w dzienniku (w milisekundach).
LogType	Określa typ wyjścia dziennika.
MaxConcurrentChecksPerPoller	Ustawia maksymalną liczbę asynchronicznych kontroli, które mogą być jednocześnie wykonywane przez każdy poller HTTP agent, poller agent lub poller SNMP.

Parametr	Opis
MaxHousekeeperDelete	Ustawia maksymalną liczbę wierszy usuwanych na zadanie w jednym cyklu porządkowania.
NodeAddress	Określa adres IP lub nazwę hosta z opcjonalnym portem, aby zastąpić sposób, w jaki frontend łączy się z serwerem.
PidFile	Określa plik PID.
ProblemHousekeepingFrequency	Ustawia częstotliwość usuwania problemów dotyczących usuniętych wyzwalaczy.
ProxyConfigFrequency	Ustawia częstotliwość wysyłania danych konfiguracji do proxy Zabbix przez serwer.
ProxyDataFrequency	Ustawia częstotliwość żądania danych historycznych z proxy Zabbix przez serwer.
ServiceManagerSyncFrequency	Ustawia częstotliwość synchronizowania konfiguracji menedżera usług.
SMSDevices	Określa pliki modemów, których może używać serwer Zabbix.
SNMPTrapperFile	Określa plik tymczasowy używany do przekazywania danych z demona pułapek SNMP do serwera.
SocketDir	Określa katalog przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.
SourceIP	Określa źródłowy adres IP.
SSHKeyLocation	Określa lokalizację kluczy publicznych i prywatnych używanych do kontroli i działań SSH.
SSLCertLocation	Określa lokalizację plików certyfikatów klienta SSL używanych do uwierzytelniania klienta.
SSLKeyLocation	Określa lokalizację plików kluczy prywatnych SSL używanych do uwierzytelniania klienta.
SSLCALocation	Określa lokalizację plików urzędu certyfikacji (CA) używanych do weryfikacji certyfikatu serwera SSL.
StartAgentPollers	Ustawia liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów agent.
StartAlerters	Ustawia liczbę wstępnie rozdzielonych instancji alerterów.
StartBrowserPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów pozycji przeglądarki.
StartConnectors	Ustawia liczbę wstępnie rozdzielonych instancji procesów roboczych konektorów.
StartDBSyncers	Ustawia liczbę wstępnie rozdzielonych instancji synchronizatorów danych historycznych.
StartDiscoverers	Ustawia liczbę wstępnie rozdzielonych instancji procesów roboczych wykrywania.
StartEscalators	Ustawia liczbę wstępnie rozdzielonych instancji eskalatorów.
StartHistoryPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów danych historycznych.
StartHTTPAgentPollers	Ustawia liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów HTTP agent.
StartHTTPPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów HTTP.
StartIPMIPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów IPMI.
StartJavaPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów Java.
StartLLDProcessors	Ustawia liczbę wstępnie rozdzielonych instancji procesów roboczych wykrywania niskopoziomowego (LLD).
StartODBCPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów ODBC.
StartPingers	Ustawia liczbę wstępnie rozdzielonych instancji pingerów ICMP.
StartPollersUnreachable	Ustawia liczbę wstępnie rozdzielonych instancji pollerów dla niedostępnych hostów (w tym IPMI i Java).
StartPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów.
StartPreprocessors	Ustawia liczbę wstępnie uruchomionych instancji procesów roboczych przetwarzania wstępnego.
StartProxyPollers	Ustawia liczbę wstępnie rozdzielonych instancji pollerów dla pasywnych proxy.
StartReportWriters	Ustawia liczbę wstępnie rozdzielonych instancji twórców raportów.
StartSNMPPollers	Ustawia liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów SNMP.
StartSNMPTrapper	Ustawia wartość „1”, aby uruchomić proces trappera SNMP.
StartTelemetryQueryPollers	Ustawia liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów zapytań telemetrycznych.
StartTimers	Ustawia liczbę wstępnie rozdzielonych instancji timerów.
StartTrappers	Ustawia liczbę wstępnie rozdzielonych instancji trapperów.
StartVMwareCollectors	Ustawia liczbę wstępnie rozdzielonych instancji kolektorów VMware.
StatsAllowedIP	Określa adresy IP lub nazwy DNS zewnętrznych instancji Zabbix, które mogą żądać statystyk.
TelemetryProvider	Ustawia źródło danych telemetrycznych.
Timeout	Określa czas oczekiwania (w sekundach) na ustanowienie połączenia i wymianę danych z proxy Zabbix, agent, usługą internetową oraz na kontrole SNMP (z wyjątkiem pozycji SNMP `walk[OID]` i `get[OID]`).
TLSCAFile	Określa pełną nazwę ścieżki pliku zawierającego certyfikaty najwyższego poziomu CA do weryfikacji certyfikatu peera, używane do szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Określa pełną nazwę ścieżki pliku zawierającego certyfikat serwera lub łańcuch certyfikatów, używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Określa ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Określa ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Parametr	Opis
TLSCipherCert	Określa ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Określa ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Określa ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Określa ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCRLFile	Określa pełną nazwę ścieżki pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSFrontendAccept	Określa poziom szyfrowania dla połączeń przychodzących z frontendu.
TLSFrontendCertIssuer	Określa dozwolonego wystawcę certyfikatu frontendu.
TLSFrontendCertSubject	Określa dozwolony podmiot certyfikatu frontendu.
TLSKeyFile	Określa pełną nazwę ścieżki pliku zawierającego prywatny klucz serwera, używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSListen	Wymaga połączeń wyłącznie TLS przez gniazdo trappera.
TmpDir	Określa katalog tymczasowy.
TrapperTimeout	Określa limit czasu trappera w sekundach dla: - pobierania danych historycznych z proxy Zabbix; - wysyłania danych konfiguracji do proxy Zabbix; - wykonywania skryptów globalnych lub zdalnych poleceń na serwerze Zabbix.
TrendCacheSize	Ustawia rozmiar pamięci podręcznej trendów.
TrendFunctionCacheSize	Ustawia rozmiar pamięci podręcznej funkcji trendów.
UnavailableDelay	Ustawia częstotliwość sprawdzania dostępności hosta w okresie niedostępności.
UnreachableDelay	Ustawia częstotliwość sprawdzania dostępności hosta w okresie nieosiągalności.
UnreachablePeriod	Ustawia, po ilu sekundach nieosiągalności host ma być traktowany jako niedostępny.
User	Obniża uprawnienia do określonego, istniejącego użytkownika w systemie.
ValueCacheSize	Ustawia rozmiar pamięci podręcznej wartości historycznych.
Vault	Określa dostawcę magazynu Vault.
VaultAppRoleID	Określa identyfikator AppRole magazynu HashiCorp Vault.
VaultAppSecretID	Określa identyfikator tajny AppRole magazynu HashiCorp Vault.
VaultDBPath	Określa lokalizację pobierania poświadczeń bazy danych według kluczy.
VaultPrefix	Określa niestandardowy prefiks ścieżki lub zapytania Vault.
VaultTLSCertFile	Określa nazwę pliku certyfikatu SSL używanego do uwierzytelniania klienta.
VaultTLSKeyFile	Określa nazwę pliku prywatnego klucza SSL używanego do uwierzytelniania klienta.
VaultToken	Określa token uwierzytelniania magazynu HashiCorp Vault.
VaultURL	Określa adres URL serwera Vault.
VMwareCacheSize	Ustawia rozmiar pamięci współdzielonej używanej do przechowywania danych VMware.
VMwareFrequency	Ustawia częstotliwość gromadzenia danych z pojedynczej usługi VMware w sekundach.
VMwarePerfFrequency	Ustawia częstotliwość pobierania statystyk liczników wydajności z pojedynczej usługi VMware w sekundach.
VMwareTimeout	Ustawia maksymalną liczbę sekund, przez które `vmware collector` będzie oczekiwać na odpowiedź usługi VMware.
WebDriverURL	Określa adres URL interfejsu WebDriver.
WebServiceURL	Określa adres URL usługi internetowej w formacie `http[s]://host:port/report`.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne demona, a nie wartości w dostarczonych plikach konfiguracji.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracji wyłącznie w kodowaniu UTF-8, bez BOM.
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Warning:
Zbyt duża liczba procesów może wyczerpać limit deskryptorów plików na proces dla menedżera przetwarzania wstępnego.

Procesy te obejmują procesy gromadzenia danych przez pollery, pollery niedostępności, pollery ODBC, pollery HTTP, pollery Java, pingery, trappery i pollery proxy, a także menedżera IPMI, trappera SNMP, procesy robocze przetwarzania wstępnego oraz procesy robocze wykrywania. Wyczerpanie limitu deskryptorów plików spowoduje **zatrzymanie** serwera Zabbix, zwykle krótko po uruchomieniu, choć czasami może to potrwać dłużej.

Aby uniknąć takich problemów, przejrzyj plik konfiguracji serwera Zabbix, aby zoptymalizować liczbę jednoczesnych kontroli i procesów. Ponadto, jeśli jest to konieczne, upewnij się, że limit deskryptorów plików jest wystarczająco wysoki, sprawdzając i dostosowując limity systemowe.

Szczegóły parametru

AlertScriptsPath

Określa lokalizację alert script. Zależy od zmiennej instalacyjnej `datadir` ustawianej w czasie kompilacji.

Domyślnie: `/usr/local/share/zabbix/alertscripts`

AllowRoot

Zezwalaj serwerowi na uruchamianie jako `root`. Jeśli opcja jest wyłączona, a serwer zostanie uruchomiony przez użytkownika `root`, serwer spróbuje zamiast tego przełączyć się na użytkownika `zabbix`. Nie ma to wpływu, jeśli serwer został uruchomiony przez zwykłego użytkownika.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

AllowSoftwareUpdateCheck

Zezwalaj interfejsowi Zabbix UI na odbieranie informacji o aktualizacjach oprogramowania z `zabbix.com`.

Domyślnie: 1
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

AllowUnsupportedDBVersions

Zezwól serwerowi na pracę z niewspieranymi wersjami bazy danych.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwól

BridgeAdapterURL

Ustaw adres URL HTTP(S) adaptera bridge Zabbix.

Przykład:

```
BridgeAdapterURL=https://bridge-adapter.example.com:10054/rpc
```

CacheSize

Ustaw rozmiar pamięci podręcznej konfiguracji (w bajtach). Rozmiar pamięci współdzielonej do przechowywania danych host, pozycja i wyzwalacz.

Domyślnie: 32M
 Zakres: 128K-64G

CacheUpdateFrequency

Ustaw częstotliwość aktualizacji pamięci podręcznej konfiguracji (w sekundach). Zobacz także opcje runtime control.

Domyślnie: 10
 Zakres: 1-3600

DBHost

Określa host bazy danych (lub katalog gniazda).

Dla MySQL:

- `localhost` lub pusty ciąg używa domyślnego gniazda UNIX-domain.

Dla PostgreSQL:

- `localhost` jest rozwiązywany przez DNS (zwykle do 127.0.0.1).
- Pusty ciąg używa domyślnego gniazda UNIX-domain.
- Ścieżka (na przykład `/var/run/pgbouncer`) powoduje użycie gniazda UNIX pod tą ścieżką.
- Lista rozdzielona przecinkami może zawierać wiele wartości `host:port`. Każdy host jest sprawdzany po kolei, aż zostanie nawiązane połączenie do odczytu i zapisu.

Przykład: `DBHost=localhost:5431,127.0.0.1:20051,zabbix.domain,[::1]:30051,[12fc::1]`

Domyślnie: `localhost`

DBName

Określ nazwę bazy danych.

Wymagane: **Tak**

DBPassword

Określ hasło bazy danych. Zakomentuj tę linię, jeśli nie jest używane hasło.

DBPort

Określ port hosta bazy danych Zabbix (zobacz DBHost).
 DBSocket i DBPort wzajemnie się wykluczają w konfiguracji serwera. Określ tylko jeden z nich albo pozostaw oba niezdefiniowane.

Domyślnie dla MySQL: 3306

Domyślnie dla PostgreSQL: 5432

Zakres: 1024-65535

DBSchema

Określ nazwę schematu bazy danych. Używane dla PostgreSQL.

DBSocket

Określ ścieżkę do pliku gniazda MySQL.
 DBSocket i DBPort wzajemnie się wykluczają w konfiguracji serwera. Określ tylko jeden z nich albo pozostaw oba niezdefiniowane.

DBUser

Określ użytkownika bazy danych.

DBTLSConnect

Określ wartość, aby wymusić połączenie TLS z bazą danych: - *required* - połącz przy użyciu TLS; - *verify_ca* - połącz przy użyciu TLS i zweryfikuj certyfikat; - *verify_full* - połącz przy użyciu TLS, zweryfikuj certyfikat oraz sprawdź, czy tożsamość bazy danych określona przez DBHost jest zgodna z jej certyfikatem.

W przypadku MySQL, od wersji 5.7.11, oraz PostgreSQL obsługiwane są następujące wartości: required, verify_ca, verify_full.
 W przypadku MariaDB, od wersji 10.2.6, obsługiwane są wartości required i verify_full.
Domyślnie nie jest ustawiona żadna opcja, a zachowanie zależy od konfiguracji bazy danych.

DBTLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu bazy danych.

Obowiązkowe: nie (tak, jeśli DBTLSConnect ma ustawioną wartość verify_ca lub verify_full)

DBTLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat serwera Zabbix do uwierzytelniania w bazie danych.

DBTLSKeyFile

Określ pełną ścieżkę do pliku zawierającego klucz prywatny używany do uwierzytelniania w bazie danych.

DBTLSCipher

Określ listę szyfrów szyfrowania, które serwer Zabbix dopuszcza dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko dla MySQL.

DBTLSCipher13

Określ listę szyfrów szyfrujących, które serwer Zabbix dopuszcza dla protokołu TLS v1.3. Obsługiwane tylko dla MySQL, począwszy od wersji 8.0.16.

DebugLevel

Ustaw poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).
 Zobacz także opcje runtime control.

Domyślnie: 3
 Zakres: 0-5

EnableGlobalScripts

Włącz globalne skrypty na serwerze Zabbix.
 Uwaga: Wykonywanie globalnych skryptów jest domyślnie włączone. Jednak w przypadku nowych instalacji rozpoczynających się od Zabbix 7.0 parametr EnableGlobalScripts jest jawnie ustawiony na 0 (wyłączone).

Domyślnie: 1
 Wartości: 0 - wyłączone; 1 - włączone

EnableMobileDevices

Włącz obsługę urządzeń mobilnych na serwerze Zabbix.

Po włączeniu serwer Zabbix będzie przetwarzać żądania dodawania lub usuwania urządzeń mobilnych oraz wysyłać powiadomienia push do tych urządzeń.

Wartość domyślna: 1
 Wartości: 0 - wyłączone; 1 - włączone

ExportDir

Określ katalog, aby włączyć eksport w czasie rzeczywistym zdarzeń, historii i trendów.

ExportFileSize

Ustaw maksymalny rozmiar pliku eksportu (w bajtach). Używane do rotacji, jeśli ustawiono `ExportDir`.

Domyślnie: `1G`
 Zakres: 1M-1G

ExportType

Określa typy encji dla eksportu w czasie rzeczywistym (zdarzenia, historia, trendy). Lista rozdzielona przecinkami. Domyślnie eksportowane są wszystkie typy. Prawidłowe tylko wtedy, gdy ustawiono `ExportDir`.
 Uwaga: jeśli `ExportType` jest określony, ale `ExportDir` nie jest, jest to błąd konfiguracji i serwer nie uruchomi się.

Przykład eksportu historii i trendów:

```
ExportType=history,trends
```

Przykład eksportu tylko zdarzeń:

```
ExportType=events
```

ExternalScripts

Określ lokalizację zewnętrznego skryptu. Zależy od zmiennej instalacyjnej `datadir` używanej podczas kompilacji.

Domyślnie: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Określ lokalizację fping6. Upewnij się, że plik binarny fping6 ma właściciela root oraz ustawiony bit SUID. Pozostaw puste (`Fping6Location=`), jeśli narzędzie fping potrafi przetwarzać adresy IPv6.

Domyślnie: `/usr/sbin/fping6`

FpingLocation

Określ lokalizację fping. Upewnij się, że plik binarny fping ma właściciela root oraz ustawiony bit SUID.

Domyślnie: `/usr/sbin/fping`

FrontendAllowedIP

Określa adresy IP lub zakresy CIDR, które mają uprawnienie do łączenia się z frontend. Lista rozdzielana przecinkami. Domyślnie wszystkie połączenia są akceptowane dla żądań frontend. Jeśli obsługa IPv6 jest włączona, to `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4.

Przykład:

```
FrontendAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

HANodeName

Określ nazwę węzła w klastrze wysokiej dostępności. Gdy pole jest puste, serwer działa w trybie autonomicznym i tworzony jest węzeł o pustej nazwie.

HistoryCacheSize

Ustaw rozmiar pamięci podręcznej historii (w bajtach). Rozmiar pamięci współdzielonej do przechowywania danych historii.

Domyślnie: `16M`
 Zakres: 128K-16G

HistoryIndexCacheSize

Ustaw rozmiar pamięci podręcznej indeksu historii (w bajtach). Rozmiar pamięci współdzielonej do indeksowania danych historii przechowywanych w pamięci podręcznej historii. Rozmiar pamięci podręcznej indeksu musi wynosić w przybliżeniu 100 bajtów na jedną pozycję.

Domyślnie: `4M`
 Zakres: 128K-16G

HistoryProvider

Ustaw dostawcę magazynu historii w formacie `<provider>[;<options>]`, gdzie:

- `<provider>` - jeden z obsługiwanych dostawców magazynu historii: `sql` (domyślnie), `elasticsearch` lub `clickhouse`.
- `<options>` - pary `key=value` rozdzielone przecinkami. Wartości mogą być ujęte w cudzysłów.

Opcje dla wszystkich dostawców magazynu historii:

- `value_types` - ustawia typy wartości do przechowywania (`uint`, `dbl`, `str`, `log`, `text`, `json`). Każdy może być użyty tylko raz i tylko dla jednego dostawcy.
- `precache` - ustaw 1 (domyślnie), aby wczytywać dane z dostawcy magazynu historii do cache wartości zbiorczo. Może to znacząco poprawić rozgrzewanie cache wartości po restarcie.

Opcje dla Elasticsearch i ClickHouse:

- `url` - ustawia adres URL HTTP lub HTTPS używany do połączenia z dostawcą magazynu historii.
- `log_slow_queries` - ustawia maksymalny czas trwania (w milisekundach), po którym zapytanie do dostawcy magazynu historii zostanie zalogowane. Opcja ta staje się dostępna od `DebugLevel=3`.
- `source_ip` - ustawia adres IP źródłowy używany dla połączeń wychodzących do dostawcy magazynu historii.

Opcje dla Elasticsearch:

- `date_index` - ustaw 1, aby włączyć przechowywanie wartości historii w różnych indeksach na podstawie daty.

Opcje dla ClickHouse:

- `db` - ustawia nazwę bazy danych.
- `username` - ustawia nazwę użytkownika używaną do logowania do bazy danych.
- `password` - ustawia hasło używane do logowania do bazy danych.
- `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_cert_location`, `ssl_key_location` - ustaw te opcje, aby skonfigurować bezpieczne połączenie (TLS/SSL) z bazą danych.

Przykład dla ClickHouse:

```
HistoryProvider=clickhouse;value_types="uint,dbl,str,log,text,json",url=http://localhost:8123,db=zabbix,us
```

Przykład dla Elasticsearch:

```
HistoryProvider=elasticsearch;value_types="uint,dbl,str,log,text,json",url=http://localhost:9200,date_inde
```

Przykład dla ClickHouse i 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

Ten parametr jest przestarzały (zamiast niego użyj HistoryProvider).
 Włącz przechowywanie wartości historii w różnych indeksach na podstawie daty. Ten parametr jest używany w konfiguracji Elasticsearch.

Domyślnie: 0
 Wartości: 0 - wyłączone; 1 - włączone

HistoryStorageURL

Ten parametr jest przestarzały (zamiast niego użyj HistoryProvider).
 Określa adres URL magazynu historii. Ten parametr jest używany podczas konfiguracji Elasticsearch.

HistoryStorageTypes

Ten parametr jest przestarzały (zamiast niego użyj HistoryProvider).
 Określa typy wartości, które mają być wysyłane do magazynu historii. Lista rozdzielana przecinkami. Ten parametr jest używany podczas konfiguracji Elasticsearch.

Domyślnie: `uint,dbl,str,log,text,json`

HousekeepingFrequency

Ustaw częstotliwość procedury housekeeping (w godzinach). Housekeeping usuwa nieaktualne informacje z bazy danych.
Uwaga: Aby zapobiec przeciążeniu housekeepera (na przykład, gdy okresy historii i trendów są znacznie skrócone), w jednym cyklu housekeeping dla każdej pozycji usuwanych jest nie więcej niż 4 razy `HousekeepingFrequency` godzin nieaktualnych informacji. Zatem jeśli `HousekeepingFrequency` wynosi 1, w jednym cyklu zostanie usunięte nie więcej niż 4 godziny nieaktualnych informacji (licząc od najstarszego wpisu).
Uwaga: Aby zmniejszyć obciążenie podczas uruchamiania serwera, housekeeping jest opóźniany o 30 minut po starcie serwera. Zatem jeśli `HousekeepingFrequency` wynosi 1, pierwsza procedura housekeeping po starcie serwera zostanie uruchomiona po 30 minutach, a następnie będzie powtarzana z godzinnym opóźnieniem.
Możliwe jest wyłączenie automatycznego housekeeping przez ustawienie `HousekeepingFrequency` na 0. W takim przypadku procedura housekeeping może być uruchomiona tylko za pomocą opcji sterowania w czasie działania `housekeeper_execute`, a okres

nieaktualnych informacji usuwanych w jednym cyklu housekeeping wynosi 4 razy okres od ostatniego cyklu housekeeping, ale nie mniej niż 4 godziny i nie więcej niż 4 dni.
Zobacz także opcje runtime control oraz szczegóły dotyczące procedury housekeeping.

Domyślnie: 1
 Zakres: 0-24

Include

Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracyjnego.
 Podczas instalacji Zabbix ten parametr jest ustawiany na `/usr/local/etc`, chyba że zmienisz to podczas kompilacji.

Ogólne zasady dla parametru `Include`:

- Ścieżka może być względna względem lokalizacji pliku `zabbix_server.conf`.
- W jednym pliku konfiguracyjnym można określić wiele parametrów `Include`.
- Zabbix przetwarza parametr `Include` natychmiast po jego odczytaniu, więc warto umieścić go na końcu pliku konfiguracyjnego.

Zasady dotyczące dołączanych plików:

- Każdy plik musi być czytelny.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie uruchomi się.

Zasady dotyczące dołączania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności dołączania (np. pliki nie są dołączane alfabetycznie).
- Nie definiuj tego samego parametru w więcej niż jednym dołączanym pliku (np. aby nadpisać ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Uważaj na kopie zapasowe tworzone automatycznie przez niektóre edytory tekstu (np. edycja `include/my_specific.conf` może utworzyć `include/my_specific.conf.BAK`), ponieważ oba pliki zostaną dołączone.

Aby dołączyć tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (`*`) do dopasowywania wzorców.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

JavaGateway

Określ adres IP lub nazwę hosta Zabbix Java gateway. Wymagane tylko wtedy, gdy uruchomione są pollery Java.

JavaGatewayPort

Określa port Zabbix Java gateway.

Domyślnie: 10052
 Zakres: 1024-32767

ListenBacklog

Ustawia maksymalną liczbę oczekujących połączeń w kolejce TCP.
Wartość domyślna jest stałą zakodowaną na stałe i zależy od systemu.
Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać bez ostrzeżenia obcięte do „maksimum określonego przez implementację”.

Domyślnie: `SOMAXCONN`
 Zakres: 0 - INT_MAX

ListenIP

Określa adres IP nasłuchiwania trappera. Lista rozdzielana przecinkami.
 Trapper będzie nasłuchiwał na wszystkich interfejsach sieciowych, jeśli ten parametr nie zostanie określony.

Domyślnie: `0.0.0.0`

ListenPort

Określ port nasłuchiwania trappera.

Domyślnie: 10051
 Zakres: 1024-32767

LoadModule

Określ moduł, który ma zostać załadowany podczas uruchamiania serwera. Moduły służą do rozszerzania funkcjonalności serwera. Moduł musi znajdować się w katalogu określonym przez `LoadModulePath` lub ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od `/`), parametr `LoadModulePath` jest ignorowany.
 Formaty:

- `LoadModule=<module.so>`
- `LoadModule=<path/module.so>`
- `LoadModule=</abs_path/module.so>`

Można określić wiele parametrów LoadModule.

LoadModulePath

Określ pełną ścieżkę do lokalizacji modułów serwera. Domyślna wartość zależy od opcji kompilacji.

LogFile

Określ plik dziennika serwera.

Obowiązkowe: Tak, jeśli LogType jest ustawione na file; w przeciwnym razie nie

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
0 - wyłącza automatyczną rotację plików dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika i rotacja pliku z jakiegokolwiek powodu się nie powiedzie, istniejący plik dziennika zostanie obcięty i utworzony ponownie.

Domyślnie: 1
 Zakres: 0-1024
 Wymagane: Tak, jeśli LogType jest ustawione na file; w przeciwnym razie nie

LogSlowQueries

Ustaw maksymalny czas trwania zapytania do bazy danych, po którym zostanie ono zapisane w logu (w milisekundach).
0 - nie loguj wolnych zapytań.
Ta opcja staje się dostępna od DebugLevel=3.

Domyślnie: 0
 Zakres: 0-3600000

LogType

Określa typ wyjścia logów: - *file* - zapisuje log do pliku określonego parametrem LogFile; - *system* - zapisuje log do syslog; - *console* - zapisuje log do standardowego wyjścia.

Domyślnie: file

MaxConcurrentChecksPerPoller

Ustaw maksymalną liczbę asynchronicznych kontroli, które mogą być wykonywane jednocześnie przez każdy poller HTTP agent, poller agent lub poller SNMP. Zobacz StartHTTPAgentPollers, StartAgentPollers oraz StartSNMPPollers.

Domyślnie: 1000
 Zakres: 1-1000

MaxHousekeeperDelete

Ustawia maksymalną liczbę wierszy usuwanych na zadanie w jednym cyklu housekeeping z tabel history, history_str, history_log, history_uint, history_text, history_bin, history_json, trends, trends_uint i problem.
Zadania housekeeping znajdują się w tabeli housekeeper w formacie: [housekeeperid], [object], [objectid].
 Ten parametr ma zastosowanie tylko do usuwania danych pozostawionych przez już usunięte pozycje.
 Jeśli ustawiono 0, nie jest stosowany żaden limit.
 W takim przypadku musisz dokładnie wiedzieć, co robisz, aby nie przeciążyć bazy danych.
 Gdy usuwanych jest wiele pozycji, zwiększa to obciążenie bazy danych, ponieważ housekeeper będzie musiał usunąć wszystkie dane historyczne, które te pozycje miały.
 Na przykład, jeśli musimy usunąć tylko 1 prototyp pozycji z szablonu, ale ten szablon jest powiązany z 50 hostami, a dla każdego hosta prototyp jest rozwijany do 100 rzeczywistych pozycji, łącznie trzeba usunąć 5000 pozycji (1*50*100).
 Jeśli dla MaxHousekeeperDelete ustawiono 500 (MaxHousekeeperDelete=500), proces housekeeper będzie musiał usunąć do 2500000 wartości (5000*500) dla usuniętych pozycji z tabel history i trends w jednym cyklu.

Zobacz też szczegóły dotyczące procedury housekeeping.

Domyślnie: 5000
 Zakres: 0-1000000

NodeAddress

Określa adres IP lub nazwę hosta z opcjonalnym portem, aby zastąpić sposób, w jaki frontend łączy się z serwerem.
Format: <address>[:<port>]

Jeśli adres IP lub nazwa hosta nie są ustawione, zostanie użyta wartość ListenIP. Jeśli ListenIP nie jest ustawione, zostanie użyta wartość localhost.
Jeśli port nie jest ustawiony, zostanie użyta wartość ListenPort. Jeśli ListenPort nie jest ustawione, zostanie użyta wartość 10051.

Ta opcja może zostać zastąpiona przez adres określony w konfiguracji frontend.

Zobacz także: parametr HANodeName; Włączanie wysokiej dostępności.

Domyślnie: localhost:10051

PidFile

Określa plik PID.

Domyślnie: /tmp/zabbix_server.pid

ProblemHousekeepingFrequency

Ustaw częstotliwość usuwania problemów dla usuniętych wyzwalaczy (w sekundach).
 Zobacz także szczegóły dotyczące procedury housekeeping.

Domyślnie: 60
 Zakres: 1-3600

ProxyConfigFrequency

Ustaw częstotliwość wysyłania danych konfiguracyjnych do proxy Zabbix przez serwer. Używane tylko dla proxy w trybie pasywnym.

Domyślnie: 10
 Zakres: 1-604800

ProxyDataFrequency

Ustawia częstotliwość pobierania danych historycznych z proxy Zabbix przez serwer. Używane tylko dla proxy w trybie pasywnym.

Domyślnie: 1
 Zakres: 1-3600

ServiceManagerSyncFrequency

Ustaw częstotliwość synchronizacji konfiguracji menedżera usług (w sekundach).

Domyślnie: 60
 Zakres: 1-3600

SMSDevices

Określ pliki modemów, które mogą być używane przez serwer Zabbix. Lista rozdzielona przecinkami.
 Wysyłanie SMS nie jest możliwe, jeśli ten parametr nie jest ustawiony.

Przykład:

```
SMSDevices=/dev/ttyUSB0,/dev/ttyUSB1
```

SNMPTrapperFile

Określ tymczasowy plik używany do przekazywania danych z demona SNMP trap do serwera.
Musi być taki sam jak w `zabbix_trap_receiver.pl` lub pliku konfiguracji SNMPTT.

Domyślnie: `/tmp/zabbix_traps.tmp`

SocketDir

Określa katalog do przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.

Domyślnie: `/tmp`

SourceIP

Określ adres IP źródłowy dla:

- połączeń wychodzących do Zabbix proxy i Zabbix agent
- połączeń bezagentowych (VMware, SSH, JMX, SNMP, Telnet i proste sprawdzenia)
- połączeń HTTP agent
- żądań HTTP JavaScript dla pozycji skryptowych
- żądań HTTP JavaScript w przetwarzaniu wstępnym
- wysyłania wiadomości e-mail z powiadomieniami (połączenia z serwerem SMTP)
- powiadomień webhook (połączenia HTTP JavaScript)
- połączeń do Vault

SSHKeyLocation

Określ lokalizację kluczy publicznych i prywatnych dla sprawdzeń SSH i działań.

SSLCertLocation

Określ lokalizację plików certyfikatu klienta SSL używanych do uwierzytelniania klienta.
 Ten parametr jest używany w monitorowaniu sieci web, do komunikacji z Vault, w przypadku dostawcy przechowywania historii ClickHouse oraz źródła danych telemetrycznych ClickHouse.

SSLKeyLocation

Określ lokalizację plików kluczy prywatnych SSL używanych do uwierzytelniania klienta.
 Ten parametr jest używany w monitorowaniu sieci web, do komunikacji z Vault, w przypadku dostawcy przechowywania historii ClickHouse oraz źródła danych telemetrycznych ClickHouse.

SSLCALocation

Określ lokalizację plików urzędu certyfikacji (CA) używanych do weryfikacji certyfikatu serwera SSL. Jeśli parametr nie zostanie ustawiony, zostanie użyty katalog systemowy.
 Należy pamiętać, że wartość tego parametru zostanie ustawiona jako opcja

libcurl CURLOPT_CAPATH. W przypadku wersji libcurl wcześniejszych niż 7.42.0 działa to tylko wtedy, gdy libcurl został skompilowany z użyciem OpenSSL. Więcej informacji można znaleźć na stronie internetowej cURL.
 Ten parametr jest używany w monitoringu sieci WWW, uwierzytelnianiu SMTP, pozycjach HTTP agent, do komunikacji z Vault, przez dostawcę przechowywania historii ClickHouse oraz przez źródło danych telemetrycznych ClickHouse.

StartAgentPollers

Ustawia liczbę wstępnie utworzonych instancji Zabbix agent pollers. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartAlerters

Ustaw liczbę wstępnie rozwidlonych instancji alerters.

Domyślnie: 3
 Zakres: 1-100

StartBrowserPollers

Ustaw liczbę wstępnie utworzonych instancji pollerów dla pozycji przeglądarki.

Domyślnie: 1
 Zakres: 0-1000

StartConnectors

Ustawia liczbę wcześniej utworzonych instancji connector workers. Proces menedżera connectora jest uruchamiany automatycznie po uruchomieniu connector worker.

Domyślnie: 0
 Zakres: 0-1000

StartDBSyncers

Ustaw liczbę wstępnie rozwidlonych instancji history syncers.
Uwaga: Zachowaj ostrożność podczas zmiany tej wartości; zwiększenie jej może przynieść więcej szkody niż pożytku. Z grubsza wartość domyślna powinna wystarczyć do obsługi do 4000 NVPS.

Domyślnie: 4
 Zakres: 1-100

StartDiscoverers

Ustaw liczbę wstępnie utworzonych instancji pracowników wykrywania[1].

Domyślnie: 5
 Zakres: 0-1000

StartEscalators

Ustawia liczbę wstępnie utworzonych instancji escalators.

Domyślnie: 1
 Zakres: 1-100

StartHistoryPollers

Ustaw liczbę wstępnie rozwidlonych instancji history pollers.
Wymagane tylko dla obliczanych kontroli.

Domyślnie: 5
 Zakres: 0-1000

StartHTTPAgentPollers

Ustaw liczbę wcześniej utworzonych instancji pollerów HTTP agent. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartHTTPPollers

Ustaw liczbę wcześniej utworzonych instancji HTTP pollers[1].

Domyślnie: 1
 Zakres: 0-1000

StartIPMIPollers

Ustaw liczbę wcześniej utworzonych instancji IPMI pollers.

Domyślnie: 0
 Zakres: 0-1000

StartJavaPollers

Ustaw liczbę wstępnie rozwidlonych instancji Java pollers[1].

Domyślnie: 0
 Zakres: 0-1000

StartLLDProcessors

Ustaw liczbę wcześniej utworzonych instancji [workerów] niskopoziomowego wykrywania (LLD) (/manual/concepts/server#server-process-types-and-threads)[1].
 Proces menedżera LLD jest uruchamiany automatycznie po uruchomieniu workera LLD.

Domyślnie: 2
 Zakres: 1-100

StartODBCPollers

Ustaw liczbę wcześniej utworzonych instancji ODBC pollers[1].

Domyślnie: 1
 Zakres: 0-1000

StartPingers

Ustaw liczbę wstępnie rozwidlonych instancji ICMP pingers[1].

Domyślnie: 1
 Zakres: 0-1000

StartPollersUnreachable

Ustaw liczbę wcześniej rozwidlonych instancji pollerów dla niedostępnych hostów (w tym IPMI i Java)[1].
 Co najmniej jeden poller dla niedostępnych hostów musi być uruchomiony, jeśli uruchomione są zwykłe pollery, pollery IPMI lub Java.

Domyślnie: 1
 Zakres: 0-1000

StartPollers

Ustaw liczbę wstępnie rozwidlonych instancji pollerów[1].

Domyślnie: 5
 Zakres: 0-1000

StartPreprocessors

Ustaw liczbę wcześniej uruchomionych instancji wątków przetwarzania wstępnego worker[1] na wartość nie mniejszą niż liczba dostępnych rdzeni CPU. Należy ustawić więcej workerów, jeśli przetwarzanie wstępne nie jest ograniczone przez CPU i obejmuje wiele żądań sieciowych.

Domyślnie: 16
 Zakres: 1-1000

StartProxyPollers

Ustaw liczbę wcześniej rozwidlonych instancji pollerów dla pasywnych proxy[1].

Domyślnie: 1
 Zakres: 0-250

StartReportWriters

Ustaw liczbę wstępnie rozwidlonych instancji report writers.
Jeśli ustawiono 0, planowane generowanie raportów jest wyłączone.
Proces report manager jest uruchamiany automatycznie po uruchomieniu report writer.

Domyślnie: 0
 Zakres: 0-100

StartSNMPPollers

Ustawia liczbę wcześniej utworzonych instancji SNMP pollers. Zobacz `MaxConcurrentChecksPerPoller`.

Domyślnie: 1
 Zakres: 0-1000

StartSNMPTrapper

Ustaw na 1, aby uruchomić proces SNMP trapper.

Domyślnie: 0
 Zakres: 0-1

StartTelemetryQueryPollers

Ustaw liczbę wstępnie utworzonych instancji asynchronicznych pollerów zapytań Telemetry. Zobacz MaxConcurrentChecksPerPoller.

Wartość domyślna: 1
 Zakres: 0-1000

StartTimers

Ustaw liczbę wcześniej utworzonych instancji timers.
Procesy timers obsługują okresy konserwacji.

Domyślnie: 1
 Zakres: 1-1000

StartTrappers

Ustawia liczbę wcześniej utworzonych instancji trapperów[1].
 Trappery akceptują połączenia przychodzące z Zabbix sender, aktywnych agentów i aktywnych proxy.

Domyślnie: 5
 Zakres: 0-1000

StartVMwareCollectors

Ustaw liczbę wstępnie rozwidlonych instancji kolektora VMware.

Domyślnie: 0
 Zakres: 0-250

StatsAllowedIP

Określ adresy IP (opcjonalnie w notacji CIDR) lub nazwy DNS zewnętrznych instancji Zabbix, które mają prawo żądać statystyk. Lista rozdzielana przecinkami.
 Jeśli ten parametr nie jest ustawiony, żadne żądania statystyk nie będą akceptowane.
 Jeśli obsługa IPv6 jest włączona, `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4.

Przykład:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

TelemetryProvider

Ustaw źródło danych telemetrycznych w formacie `<provider>[;<options>]`, gdzie:

- `<provider>` - jedno z obsługiwanych źródeł danych telemetrycznych: `clickhouse`
- `<options>` - rozdzielane przecinkami pary `key=value`. Wartości można ujmować w cudzysłowy.

Opcje dla ClickHouse:

- `url` - ustaw adres URL HTTP lub HTTPS używany do łączenia ze źródłem danych telemetrycznych.
- `db` - ustaw nazwę bazy danych.
- `username` - ustaw nazwę użytkownika używaną do logowania do bazy danych.
- `password` - ustaw hasło używane do logowania do bazy danych.
- `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_cert_location`, `ssl_key_location` - ustaw te opcje, aby skonfigurować bezpieczne połączenie (TLS/SSL) z bazą danych.
- `source_ip` - ustaw źródłowy adres IP używany dla połączeń wychodzących ze źródłem danych telemetrycznych.
- `vault_path` - ustaw ścieżkę do sekretu Vault zawierającego wartości `username` i `password`. Nie używaj razem z opcjami `username` i `password`.

Przykład dla ClickHouse:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,username=zabbix,password=zabbix
```

Przykład dla ClickHouse z Vault:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,vault_path=clickhouse
```

Timeout

Określa, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy Zabbix, agent, usługą webową oraz starszymi sprawdzeniami SNMP (pojedynczy numer OID lub ciąg znaków).

Ten parametr definiuje czas trwania różnych operacji komunikacyjnych:

- zdalne wykonywanie poleceń na agencie Zabbix
- wykonywanie poleceń SSH/Telnet
- żądania do usługi webowej Zabbix
- limit czasu komunikacji dla żądania testowego mediatype oraz metody `history.push`
- ponowne planowanie pozycji, gdy interfejs IPMI staje się niedostępny
- wysyłanie odpowiedzi do proxy Zabbix w przypadku niepowodzenia wymiany danych z powodu problemów z uprawnieniami lub szyfrowaniem
- termin dla asynchronicznych gniazd IPC oraz opcji sterowania w czasie działania
- połączenia JMX
- pobieranie statystyk z zdalnego proxy Zabbix lub serwera
- wysyłanie odpowiedzi do frontend Zabbix
- asynchroniczne żądania DNS pollera
- odpowiedź dla heartbeat aktywnej kontroli
- pobieranie danych (wartości) agenta Zabbix z aktywnych agentów
- pobieranie danych z Zabbix sender
- odpowiedź, gdy trapper nie może przetworzyć żądania
- wysyłanie listy aktywnych kontroli do agenta Zabbix

Ten limit czasu **nie** będzie używany dla tych kontroli, dla których w frontend skonfigurowano ustawienia flexible timeout (na poziomie globalnym, proxy lub dla pojedynczej pozycji). Na przykład pozycje SNMP `walk[OID]` i `get[OID]` używają limitu czasu skonfigurowanego w frontend; starsze sprawdzenia SNMP nadal używają wartości limitu czasu serwera.

Domyślnie: 3
 Zakres: 1-30

TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używaną do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSCipherAll

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

TLSCipherPSK

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCRLFile

Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSFrontendAccept

Określ poziom szyfrowania dla połączeń przychodzących z frontend.

Można podać wiele wartości, oddzielając je przecinkiem:

- unencrypted - akceptuj połączenia bez szyfrowania.
- cert - akceptuj połączenia zabezpieczone za pomocą TLS i certyfikatu.

Domyślnie: `unencrypted`

TLSFrontendCertIssuer

Określ dozwolonego wystawcę certyfikatu frontend.

TLSFrontendCertSubject

Określ dozwolony subject certyfikatu frontend.

TLSKeyFile

Podaj pełną ścieżkę do pliku zawierającego prywatny klucz serwera, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSListen

Wymagaj połączeń wyłącznie TLS na gnieździe trapper.

Obsługiwane wartości:

- required - akceptuj tylko połączenia TLS

TmpDir

Określ katalog tymczasowy.

Domyślnie: `/tmp`

TrapperTimeout

Określ limit czasu trappera w sekundach dla:

- pobierania danych historycznych z proxy Zabbix
- wysyłania danych konfiguracyjnych do proxy Zabbix
- wykonywania globalnego skryptu lub zdalnego polecenia na serwerze Zabbix

Domyślnie: 300
 Zakres: 1-300

TrendCacheSize

Ustaw rozmiar pamięci podręcznej trendów w bajtach.
Rozmiar pamięci współdzielonej do przechowywania trendów.

Domyślnie: `4M`
 Zakres: 128K-16G

TrendFunctionCacheSize

Ustaw rozmiar pamięci podręcznej funkcji trendu w bajtach.
Rozmiar pamięci współdzielonej do buforowania obliczonych danych funkcji trendu.

Domyślnie: `4M`
 Zakres: 128K-2G

UnavailableDelay

Ustawia częstotliwość sprawdzania dostępności hosta podczas okresu niedostępności w sekundach.

Domyślnie: 60
 Zakres: 1-3600

UnreachableDelay

Ustaw częstotliwość sprawdzania dostępności hosta w okresie niedostępności w sekundach.

Domyślnie: 15
 Zakres: 1-3600

UnreachablePeriod

Ustawia, po ilu sekundach niedostępności uznać host za niedostępny.

Domyślnie: 45
 Zakres: 1-3600

Użytkownik

Obniż uprawnienia do określonego, istniejącego użytkownika w systemie.
Ma to efekt tylko wtedy, gdy jest uruchamiane jako 'root' i AllowRoot jest wyłączone.

Domyślnie: zabbix

ValueCacheSize

Ustaw rozmiar pamięci podręcznej wartości historii (w bajtach).
 Rozmiar pamięci współdzielonej używanej do buforowania żądań dotyczących danych historycznych pozycji.
 Ustawienie wartości 0 wyłącza pamięć podręczną wartości (niezalecane).
 Gdy w pamięci podręcznej wartości zabraknie pamięci współdzielonej, co 5 minut w dzienniku serwera zapisywany jest komunikat ostrzegawczy.

Wartość domyślna: 8M
 Zakres: 0, 128K-64G

Vault

Określ dostawcę vault: - *HashiCorp* - HashiCorp KV Secrets Engine w wersji 2; - *CyberArk* - CyberArk Central Credential Provider
Musi być zgodny z dostawcą vault ustawionym we frontendzie.

Domyślnie: HashiCorp

VaultAppRoleID

Określ identyfikator AppRole usługi HashiCorp Vault.
 Jeśli zostanie ustawiony, włącza uwierzytelnianie AppRole. Nie może być używany razem z VaultToken.
 Obsługiwane tylko dla HashiCorp Vault.

VaultAppSecretID

Określ secret ID AppRole w HashiCorp Vault.
 Obsługiwane tylko dla HashiCorp Vault.

VaultDBPath

Określa lokalizację pobierania poświadczeń bazy danych według kluczy. Jest to ścieżka lub zapytanie Vault, zależnie od używanego Vault.

Klucze używane przez HashiCorp to password i username.

Przykładowa ścieżka z VaultPrefix=/v1/secret/data/zabbix/:

```
database
```

Przykładowa ścieżka bez VaultPrefix:

```
secret/zabbix/database
```

Klucze używane przez CyberArk to Content i UserName.

Przykład:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_server_database
```

Tej opcji można użyć tylko wtedy, gdy DBUser i DBPassword nie są określone.

VaultPrefix

Określ niestandardowy prefiks dla ścieżki lub zapytania Vault, w zależności od Vault. Jeśli nie zostanie określony, zostaną użyte najbardziej odpowiednie wartości domyślne.
 Należy pamiętać, że data jest automatycznie dodawane po punkcie montowania dla HashiCorp, jeśli VaultPrefix nie jest określony.

Przykładowy prefiks dla HashiCorp:

```
/v1/secret/data/zabbix/
```

Przykładowy prefiks dla Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Określ nazwę pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM1.
 Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole pliku klucza SSL puste.
 Katalog zawierający ten plik jest określany przez parametr konfiguracji SSLCertLocation.
 Ta opcja może zostać pominięta, ale jest zalecana dla sejfu CyberArkCCP.

VaultTLSKeyFile

Określ nazwę pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik prywatnego klucza musi być w formacie PEM1.
Katalog zawierający ten plik jest określany przez parametr konfiguracji SSLKeyLocation.
Opcję tę można pominąć, ale jest zalecana dla sejfu CyberArkCCP.

VaultToken

Określ token uwierzytelniania HashiCorp Vault. Token uwierzytelniania HashiCorp Vault powinien zostać wygenerowany wyłącznie dla serwer Zabbix z uprawnieniami tylko do odczytu do ścieżek określonych w makrach Vault oraz z uprawnieniami tylko do odczytu do ścieżki określonej w opcjonalnym parametrze konfiguracyjnym VaultDBPath.
 Występuje błąd, jeśli `VaultToken` i zmienna środowiskowa `VAULT_TOKEN` są zdefiniowane jednocześnie.
 Nie można używać z VaultAppRoleID.
 Obsługiwane tylko dla HashiCorp Vault.

VaultURL

Określ adres URL serwera vault. Katalog certyfikatów CA obowiązujących w całym systemie zostanie użyty, jeśli nie zostanie określony SSLCALocation.

Domyślnie: `https://127.0.0.1:8200`

VMwareCacheSize

Ustaw rozmiar pamięci współdzielonej do przechowywania danych VMware. Do monitorowania użycia pamięci podręcznej VMware można użyć wewnętrznego sprawdzenia `zabbix[vmware,buffer,...]` (zobacz Wewnętrzne sprawdzenia).
Należy pamiętać, że pamięć współdzielona nie jest przydzielana, jeśli nie skonfigurowano żadnych instancji `vmware collector` do uruchomienia.

Domyślnie: 8M
 Zakres: 256K-2G

VMwareFrequency

Ustaw częstotliwość zbierania danych z pojedynczej usługi VMware (w sekundach). Ta częstotliwość powinna być ustawiona na najmniejszy interwał aktualizacji dowolnej pozycji monitorującej VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwarePerfFrequency

Ustaw częstotliwość pobierania statystyk liczników wydajności z pojedynczej usługi VMware (w sekundach).
Ta częstotliwość powinna być ustawiona na najmniejszy interwał aktualizacji dowolnej monitorowanej pozycji VMware pozycja, która używa liczników wydajności VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwareTimeout

Ustaw maksymalną liczbę sekund, przez którą `vmware collector` będzie czekać na odpowiedź z usługi VMware (vCenter lub hipernadzorcy ESX).

Domyślnie: 10
 Zakres: 1-300

WebDriverURL

Określ adres URL interfejsu WebDriver.

Przykład (używany z samodzielnym serwerem Selenium WebDriver):

```
WebDriverURL=http://localhost:4444
```

WebServiceURL

Określ adres URL usługi sieciowej w formacie `http[s]://host:port/report`.

Przykład:

```
WebServiceURL=http://localhost:10053/report
```

Schemat `http://` można pominąć tylko w przypadku połączeń bez TLS (HTTP); przy skonfigurowanym TLS należy użyć `https://`.

Przypisy

[1] Należy pamiętać, że zbyt duża liczba procesów zbierających dane (pollers, unreachable pollers, ODBC pollers, HTTP pollers, Java pollers, pingers, trappers, proxypollers) wraz z IPMI manager, SNMP trapper, preprocessing workers i discovery workers może wyczerpać limit deskryptorów plików na proces dla preprocessing manager.

Warning:
Wyczerpanie limitu deskryptorów plików spowoduje zatrzymanie serwera Zabbix, zwykle krótko po uruchomieniu, ale czasami dopiero po dłuższym czasie. Aby uniknąć takich problemów, przejrzyj plik konfiguracyjny serwera Zabbix, aby zoptymalizować liczbę równoczesnych kontroli i procesów. Dodatkowo, jeśli to konieczne, upewnij się, że limit deskryptorów plików jest ustawiony wystarczająco wysoko, sprawdzając i dostosowując limity systemowe.

Proxy

Przegląd proxy Zabbix to proces, który może zbierać dane monitorowania z jednego lub większej liczby monitorowanych urządzeń i wysyłać te informacje do serwera Zabbix, działając zasadniczo w jego imieniu. Wszystkie zebrane dane są buforowane lokalnie, a następnie przesyłane do serwera Zabbix, do którego należy proxy.

Wdrożenie proxy jest opcjonalne, ale może przynieść znaczne korzyści, ponieważ umożliwia rozłożenie obciążenia pojedynczego serwera Zabbix. Jeśli dane są zbierane wyłącznie przez proxy, przetwarzanie po stronie serwera wymaga mniej zasobów procesora i operacji wejścia/wyjścia dysku.

proxy Zabbix jest idealnym rozwiązaniem do scentralizowanego monitorowania zdalnych lokalizacji, oddziałów i sieci, w których nie ma lokalnych administratorów.

proxy Zabbix wymaga oddzielnej bazy danych.

Attention:
Należy pamiętać, że bazami danych obsługiwanymi przez proxy Zabbix są SQLite, MySQL i PostgreSQL.

Zobacz także:

- Monitorowanie rozproszone z użyciem proxy
- Równoważenie obciążenia proxy i wysoka dostępność
- Synchronizacja konfiguracji monitorowania
- Protokół wymiany danych między serwerem a proxy
- Parametry konfiguracji: proxy Zabbix
- Konfiguracja proxy Zabbix Cloud

Uruchamianie proxy Jeśli zainstalowano jako pakiet

Zabbix proxy działa jako proces demona. Proxy można uruchomić, wykonując:

```
systemctl start zabbix-proxy
```

To zadziała w większości systemów GNU/Linux. W innych systemach może być konieczne uruchomienie:

```
/etc/init.d/zabbix-proxy start
```

Podobnie, aby zatrzymać/uruchomić ponownie/wyświetlić status Zabbix proxy, użyj następujących poleceń:

```
systemctl stop zabbix-proxy
systemctl restart zabbix-proxy
systemctl status zabbix-proxy
```

Uruchamianie ręczne

Jeśli powyższe nie działa, musisz uruchomić go ręcznie. Znajdź ścieżkę do pliku binarnego `zabbix_proxy` i wykonaj:

```
zabbix_proxy
```

Możesz używać następujących parametrów wiersza poleceń z proxy Zabbix:

Parameter	Description
`-c --config <config-file>`	Ścieżka do pliku konfiguracyjnego.
`-f --foreground`	Uruchom proxy Zabbix na pierwszym planie.
`-R --runtime-control <option>`	Wykonaj funkcje administracyjne.
`-T --test-config`	Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.
`-h --help`	Wyświetl informacje pomocy i zakończ działanie.

Parameter	Description
`-V --version`	Wyświetl numer wersji.

Przykłady uruchamiania proxy Zabbix z parametrami wiersza poleceń:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf
zabbix_proxy --help
zabbix_proxy -V
```

Sterowanie w czasie działania

Opcje sterowania w czasie działania:

Option	Description	Target
`config_cache_reload`	Przeładuj pamięć podręczną konfiguracji. Ignorowane, jeśli pamięć podręczna jest obecnie ładowana. Aktywny proxy Zabbix połączy się z serwerem Zabbix i zażąda danych konfiguracyjnych. Pasywny proxy Zabbix zażąda danych konfiguracyjnych od serwera Zabbix przy następnym połączeniu serwera z proxy.	
`history_cache_clear[=<target>]`	Wyczyść pamięć podręczną historii dla pozycji określonej przez jej ID. Dotyczy wszystkich wartości pozycji, z wyjątkiem pierwszej i ostatniej wartości.	**target** - ID pozycji.
`diaginfo[=<section>]`	Zbierz informacje diagnostyczne w pliku dziennika proxy.	`historycache` - statystyki pamięci podręcznej historii; `preprocessing` - statystyki menedżera preprocessingu; `locks` - lista mutexów (jest pusta w systemach *BSD*).
`snmp_cache_reload`	Przeładuj pamięć podręczną SNMP - wyczyść właściwości silnika SNMP (engine time, engine boots, engine id, credentials) dla wszystkich hostów. Użyj, aby wymusić globalne wyczyszczenie pamięci podręcznej podczas rozwiązywania problemów z SNMP.	
`housekeeper_execute`	Uruchom procedurę housekeeping. Ignorowane, jeśli procedura housekeeping jest obecnie w toku.	
`log_level_increase[=<target>]`	Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony. Nieobsługiwane w systemach *BSD*.	**process type** - wszystkie procesy określonego typu (np. `poller`). Zobacz wszystkie typy procesów proxy. **process type,N** - typ procesu i numer (np. `poller,3`) **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'.
`log_level_decrease[=<target>]`	Obniż poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony. Nieobsługiwane w systemach *BSD*.	

Option	Description	Target
`prof_enable[=<target>]`	Włącz profilowanie. Wpływa na wszystkie procesy, jeśli target nie jest określony. Włączone profilowanie dostarcza szczegółów wszystkich rwlocks/mutexes według nazwy funkcji.	**process type** - wszystkie procesy określonego typu (np. `history syncer`). Zobacz wszystkie typy procesów proxy. **process type,N** - typ procesu i numer (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'. **scope** - `rwlock`, `mutex`, `processing` mogą być użyte z typem procesu i numerem (np. `history syncer,1,processing`) lub ze wszystkimi procesami danego typu (np. `history syncer,rwlock`).
`prof_disable[=<target>]`	Wyłącz profilowanie. Wpływa na wszystkie procesy, jeśli target nie jest określony.	**process type** - wszystkie procesy określonego typu (np. `history syncer`). Zobacz wszystkie typy procesów proxy. **process type,N** - typ procesu i numer (np. `history syncer,1`). **pid** - identyfikator procesu (1 do 65535). Dla większych wartości określ target jako 'process type,N'.

Gdy typ procesu jest określony jako target sterowania w czasie działania, polecenie ma zastosowanie tylko do uruchomionych procesów tego typu. Jeśli żaden proces określonego typu nie jest obecnie uruchomiony, polecenie zwraca błąd:

```
zabbix_proxy -R log_level_increase='data sender'
Cannot redirect signal: "data sender" process does not exist
```

Przykład użycia sterowania w czasie działania do przeładowania pamięci podręcznej konfiguracji proxy:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R config_cache_reload
```

Przykład użycia sterowania w czasie działania do wyczyszczenia pamięci podręcznej historii dla pozycji:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R history_cache_clear=42243
```

Przykłady użycia sterowania w czasie działania do zebrania informacji diagnostycznych:

```
### Zbierz wszystkie dostępne informacje diagnostyczne w pliku dziennika proxy:
zabbix_proxy -R diaginfo

### Zbierz statystyki pamięci podręcznej historii w pliku dziennika proxy:
zabbix_proxy -R diaginfo=historycache
```

Przykład użycia sterowania w czasie działania do przeładowania pamięci podręcznej SNMP:

```
zabbix_proxy -R snmp_cache_reload
```

Attention:
Gdy interfejs SNMPv3 jest aktualizowany za pomocą interfejsu Zabbix UI, Zabbix w większości przypadków automatycznie przeładuje nowe poświadczenia SNMPv3 dla tego interfejsu; używaj `-R snmp_cache_reload` tylko wtedy, gdy odpytywanie nadal kończy się niepowodzeniem po zmianie poświadczeń (na przykład z powodu niespójności engineBoots/engineID lub urządzeń niezgodnych z RFC), albo gdy chcesz wymusić globalne wyczyszczenie pamięci podręcznej SNMP w celu rozwiązywania problemów.

Przykład użycia sterowania w czasie działania do uruchomienia housekeepera:

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R housekeeper_execute
```

Przykłady użycia sterowania w czasie działania do zmiany poziomu logowania:

```
### Zwiększ poziom logowania wszystkich procesów:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase

### Zwiększ poziom logowania drugiego procesu poller:
```

```
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=poller,2

### Zwiększ poziom logowania procesu o PID 1234:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_increase=1234

### Zmniejsz poziom logowania wszystkich procesów http poller:
zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R log_level_decrease="http poller"
```

Uruchamianie przez użytkownika

proxy jest przeznaczony do uruchamiania jako użytkownik inny niż root. Będzie działać jako dowolny użytkownik inny niż root, z którego uprawnieniami został uruchomiony. Dlatego można uruchomić proxy jako dowolnego użytkownika innego niż root bez żadnych problemów.

Jeśli spróbujesz uruchomić go jako „root", przełączy się na zakodowanego na stałe użytkownika „zabbix", który musi istnieć w systemie. Możesz uruchomić proxy jako „root" tylko po odpowiedniej modyfikacji parametru `AllowRoot` w pliku konfiguracyjnym proxy.

Plik konfiguracyjny

Domyślna ścieżka do pliku konfiguracyjnego zależy od systemu operacyjnego i metody instalacji Zabbix:

- W przypadku instalacji ze źródeł: `/usr/local/etc/zabbix_proxy.conf` lub zgodnie z ustawionymi podczas kompilacji wartościami zmiennych `--sysconfdir` albo `--prefix`.
- W przypadku instalacji z pakietów: `/etc/zabbix/zabbix_proxy.conf`.

Szczegółowe informacje o konfigurowaniu `zabbix_proxy` znajdują się w opisie opcji pliku konfiguracyjnego.

Typy procesów i wątków proxy

- `agent poller` - asynchroniczny proces odpytywania dla pasywnych kontroli z wątkiem roboczym;
- `apm manager` - proces zarządzający monitorowaniem wydajności aplikacji (APM);
- `apm worker` - proces obsługujący zadania monitorowania wydajności aplikacji (APM) z procesu `apm manager`;
- `availability manager` - proces aktualizujący dostępność hostów;
- `browser poller` - proces odpytywania dla kontroli pozycji przeglądarki;
- `configuration syncer` - proces zarządzający pamięci podręcznej danych konfiguracyjnych;
- `data sender` - proces wysyłający dane proxy;
- `discovery manager` - proces zarządzający wykrywaniem urządzeń;
- `discovery worker` - proces obsługujący zadania wykrywania z procesu discovery manager;
- `history syncer` - proces zapisujący dane w bazie danych historii;
- `housekeeper` - proces usuwający nieaktualną historię pozycji;
- `http agent poller` - asynchroniczny proces odpytywania dla kontroli HTTP z wątkiem roboczym;
- `http poller` - proces odpytywania monitorowania sieci WWW;
- `icmp pinger` - proces odpytywania dla kontroli icmpping;
- `internal poller` - proces odpytywania dla kontroli wewnętrznych;
- `ipmi manager` - proces zarządzający odpytywaniem IPMI;
- `ipmi poller` - proces odpytywania dla kontroli IPMI;
- `java poller` - proces odpytywania dla kontroli Java;
- `odbc poller` - proces odpytywania dla kontroli ODBC;
- `poller` - standardowy proces odpytywania dla pasywnych kontroli;
- `preprocessing manager` - proces zarządzający zadaniami wstępnego przetwarzania z wątkami roboczymi wstępnego przetwarzania;
- `preprocessing worker` - wątek wstępnego przetwarzania danych;
- `self-monitoring` - proces zbierający wewnętrzne statystyki serwera;
- `snmp poller` - asynchroniczny proces odpytywania dla kontroli SNMP z wątkiem roboczym (tylko pozycje `walk[OID]` i `get[OID]`);
- `snmp trapper` - proces trapper dla pułapek SNMP;
- `supervisor` - koordynuje uruchamianie i zatrzymywanie procesów oraz wątków serwera, a także monitoruje ich gotowość;
- `task manager` - proces zdalnego wykonywania zadań żądanych przez inne komponenty (np. zamykanie problemu, potwierdzanie problemu, natychmiastowe sprawdzanie wartości pozycji, funkcjonalność zdalnych poleceń);
- `telemetry query poller` - proces odpytywania dla kontroli pozycji zapytań telemetrycznych;
- `trapper` - proces trapper dla kontroli aktywnych, pułapek i komunikacji z proxy;
- `unreachable poller` - proces odpytywania dla niedostępnych urządzeń;
- `vmware collector` - moduł zbierający dane VMware odpowiedzialny za gromadzenie danych z usług VMware.

Plik dziennika proxy może służyć do obserwowania tych typów procesów.

Plik dziennika proxy jest tworzony z uprawnieniami odczytu i zapisu wyłącznie dla właściciela pliku. Ponadto plik jest dostępny do odczytu dla grupy właściciela. Wszystkie pozostałe uprawnienia są odrzucane.

Różne typy procesów proxy Zabbix można monitorować za pomocą wewnętrznej pozycji `zabbix[process,<type>,<mode>,<state>]`.

Statystyki transakcji synchronizatora historii

Tytuł procesu synchronizatora historii wyświetla szczegółowe statystyki dotyczące transakcji synchronizatora historii.

```
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
```

Czasy w formacie `processed...in N (<timings>) sec` oznaczają:

- Czas spędzony na zapisywaniu wartości pozycji do bazy danych.
- Czas spędzony na aktualizowaniu danych pozycji (stan, błędy).

Procedura housekeeping

Proxy Zabbix ma proces `housekeeper`, który usuwa przestarzałą historię i trendy pozycji. Działa on w cyklach, a częstotliwość jest określana przez `HousekeepingFrequency`, natomiast limit usuwania w jednym cyklu jest określany przez `ProxyLocalBuffer` i `ProxyOfflineBuffer`. W przeciwieństwie do procedury housekeeping serwera Zabbix, proces `housekeeper` proxy nie korzysta z tabeli `housekeeper` — usuwa wszystkie przestarzałe dane raz na każdy cykl housekeeping.

Obsługiwane platformy Proxy Zabbix działa na tej samej liście obsługiwanych platform co serwer Zabbix.

Bufor pamięci Bufor pamięci umożliwia przechowywanie nowych danych (wartości pozycji, wykrywanie sieci, automatyczna rejestracja hostów) w buforze i przesyłanie ich do serwera Zabbix bez dostępu do bazy danych. Bufor pamięci został wprowadzony dla proxy w Zabbix 7.0.

W instalacjach starszych niż Zabbix 7.0 zebrane dane były przechowywane w bazie danych przed przesłaniem ich do serwera Zabbix. W przypadku tych instalacji po uaktualnieniu do Zabbix 7.0 nadal jest to domyślne zachowanie.

W celu optymalizacji wydajności zaleca się skonfigurowanie używania bufora pamięci na proxy. Można to zrobić, zmieniając wartość `ProxyBufferMode` z `disk` (zakodowana na stałe wartość domyślna dla istniejących instalacji) na `hybrid` (zalecane) lub `memory`. Wymagane jest również ustawienie rozmiaru bufora pamięci (parametr `ProxyMemoryBufferSize`).

W trybie hybrydowym bufor jest chroniony przed utratą danych przez opróżnianie nieprzesłanych danych do bazy danych, jeśli proxy zostanie zatrzymane, bufor będzie pełny lub dane będą zbyt stare. Po opróżnieniu wszystkich wartości do bazy danych proxy ponownie zaczyna używać bufora pamięci.

W trybie pamięci używany jest bufor pamięci, jednak nie ma ochrony przed utratą danych. Jeśli proxy zostanie zatrzymane lub pamięć zostanie przepełniona, nieprzesłane dane zostaną odrzucone.

Tryb hybrydowy (`ProxyBufferMode=hybrid`) jest stosowany we wszystkich nowych instalacjach od wersji Zabbix 7.0.

Dodatkowe parametry, takie jak `ProxyMemoryBufferSize` i `ProxyMemoryBufferAge`, określają odpowiednio rozmiar bufora pamięci i maksymalny wiek danych w buforze.

Należy pamiętać, że w przypadku sprzecznej konfiguracji proxy wyświetli błąd i nie uruchomi się, na przykład gdy:

- `ProxyBufferMode` jest ustawiony na `hybrid` lub `memory`, a `ProxyMemoryBufferSize` ma wartość 0.
- `ProxyBufferMode` jest ustawiony na `hybrid` lub `memory`, a `ProxyLocalBuffer` nie ma wartości 0.

Ustawienia regionalne Należy pamiętać, że proxy wymaga ustawień regionalnych UTF-8, aby niektóre tekstowe pozycje mogły być poprawnie interpretowane. Większość nowoczesnych systemów typu Unix ma domyślnie ustawione ustawienia regionalne UTF-8, jednak w niektórych systemach może być konieczne ich ręczne skonfigurowanie.

Obliczanie kolejek podczas konserwacji

> **Attention:**
> Zabbix proxy nie jest świadomy okresów konserwacji; szczegółowe informacje można znal
> podczas konserwacji.

Korzystanie z frontend Zabbix za proxy Jeśli frontend Zabbix działa za serwerem proxy, ścieżkę cookie w pliku konfiguracji proxy należy przepisać tak, aby odpowiadała ścieżce używanej przez reverse proxy. Zobacz przykłady poniżej. Jeśli ścieżka cookie nie zostanie przepisana, użytkownicy mogą napotkać problemy z autoryzacją podczas próby zalogowania się do frontend Zabbix.

Przykładowa konfiguracja NGINX

```
### ..
location / {
### ..
proxy_cookie_path /zabbix /;
proxy_pass http://192.168.0.94/zabbix/;
### ..
```

Przykładowa konfiguracja Apache

```
### ..
ProxyPass "/" http://host/zabbix/
ProxyPassReverse "/" http://host/zabbix/
ProxyPassReverseCookiePath /zabbix /
ProxyPassReverseCookieDomain host zabbix.example.com
### ..
```

Monitorowanie rozproszone z użyciem proxy

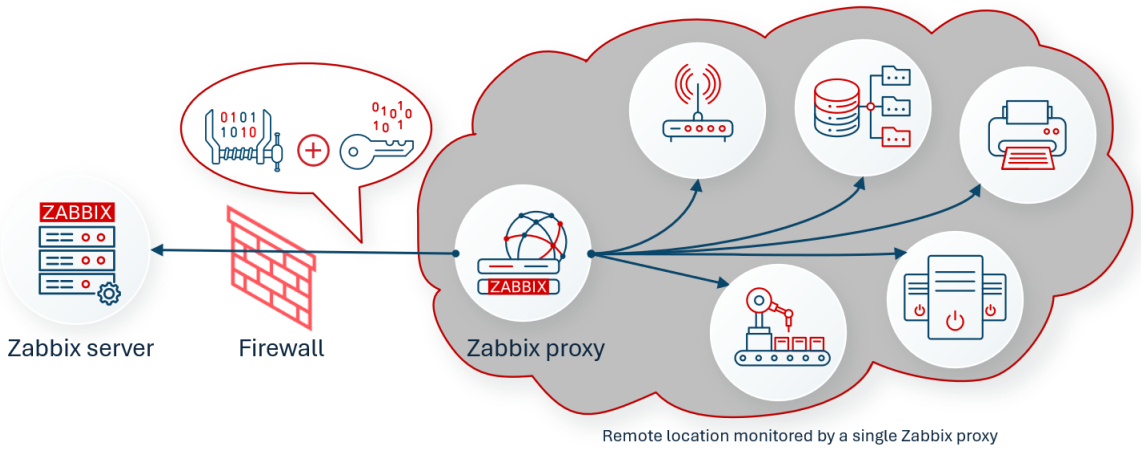
Przegląd

Proxy Zabbix można wykorzystać do utworzenia skutecznego i niezawodnego rozproszonego systemu monitorowania.

Proxy są wdrażane lokalnie w celu zbierania danych dotyczących wydajności i dostępności. Następnie dane są przekazywane do scentralizowanego serwera.

Proxy Zabbix można wykorzystać do:

- Monitorowania zdalnych lokalizacji
- Monitorowania lokalizacji, w których komunikacja jest zawodna
- Odciążenia serwera Zabbix podczas monitorowania tysięcy urządzeń
- Zbierania danych telemetrycznych na potrzeby monitorowania wydajności aplikacji (APM)



Proxy wymaga tylko jednego połączenia TCP z serwerem Zabbix. Dzięki temu łatwiej ominąć zaporę sieciową, ponieważ wystarczy skonfigurować jedną regułę zapory.

Attention:
Proxy Zabbix musi korzystać z oddzielnej bazy danych. Wskazanie bazy danych serwera Zabbix spowoduje uszkodzenie konfiguracji.

Wszystkie dane zebrane przez proxy są przechowywane lokalnie przed przesłaniem ich do serwera. Dzięki temu dane nie zostaną utracone z powodu tymczasowych problemów z komunikacją z serwerem. Parametry `ProxyLocalBuffer` i `ProxyOfflineBuffer` w pliku konfiguracyjnym proxy określają, jak długo dane są przechowywane lokalnie.

Attention:
Może się zdarzyć, że proxy, które otrzymuje najnowsze zmiany konfiguracji bezpośrednio z bazy danych serwera Zabbix, będzie mieć bardziej aktualną konfigurację niż serwer Zabbix, którego konfiguracja może nie być aktualizowana równie szybko z powodu wartości parametru CacheUpdateFrequency. W rezultacie proxy może rozpocząć zbieranie danych i wysyłać je do serwera Zabbix, który będzie te dane ignorować.

Proxy Zabbix jest kolektorem danych. Nie oblicza wyzwalaczy, nie przetwarza zdarzeń ani nie wysyła alertów. Przegląd funkcji obsługiwanych przez proxy przedstawiono w poniższej tabeli:

Function		Supported by proxy
Items		
	Zabbix agent checks	**Yes**
	Zabbix agent checks (active)	**Yes** [1]
	Simple checks	**Yes**
	Trapper items	**Yes**
	SNMP checks	**Yes**
	SNMP traps	**Yes**
	IPMI checks	**Yes**
	JMX checks	**Yes**
	Log file monitoring	**Yes**
	Internal checks	**Yes**
	SSH checks	**Yes**
	Telnet checks	**Yes**
	External checks	**Yes**
	Dependent items	**Yes**
	Script items	**Yes**
	Browser items	**Yes**
	Telemetry query items	**Yes**
Built-in web monitoring		**Yes**
Item value preprocessing		**Yes**
Network discovery		**Yes**
Active agent autoregistration		**Yes**
Low-level discovery		**Yes** [2]
Remote commands		**Yes**
Telemetry data collection (APM)		**Yes** [3]
Calculating triggers		*No*
Processing events		*No*
Event correlation		*No*
Sending alerts		*No*

Note:
[1] Aby upewnić się, że agent pyta proxy (a nie serwer) o aktywne sprawdzenia, proxy musi być wymienione w parametrze **ServerActive** w pliku konfiguracyjnym agenta.
 [2] W przypadku LLD proxy Zabbix tylko zbiera i wstępnie przetwarza dane, a następnie wysyła je do serwera Zabbix w celu dalszego przetwarzania.
 [3] Dane telemetryczne są zbierane tylko wtedy, gdy APM jest włączone na proxy (zobacz kartę *APM* w konfiguracji proxy); serwer Zabbix nie może odbierać danych telemetrycznych.

Funkcje proxy

Przy podejmowaniu decyzji o użyciu lub nieużywaniu proxy należy wziąć pod uwagę kilka kwestii.

	Proxy
Lekki	**Tak**
GUI	Nie
Działa niezależnie	**Tak**
Łatwa konserwacja	**Tak**
Automatyczne tworzenie bazy danych	**Tak**[1]
Lokalne administrowanie	Nie
Gotowy do pracy na sprzęcie wbudowanym	**Tak**
Jednokierunkowe połączenia TCP	**Tak**
Scentralizowana konfiguracja	**Tak**
Generuje powiadomienia	Nie

[1] Funkcja automatycznego tworzenia bazy danych działa tylko z SQLite. Inne obsługiwane bazy danych wymagają ręcznej konfiguracji.

Attention:
Proxy Zabbixa nie uwzględnia okresów konserwacji; szczegóły można znaleźć w sekcji Obliczanie kolejek podczas konserwacji.

Ochrona przed przeciążeniem

Jeśli serwer Zabbix był przez pewien czas niedostępny, a proxy zebrały dużo danych, to po uruchomieniu serwera może dojść do jego przeciążenia (użycie pamięci podręcznej historii utrzymuje się przez pewien czas na poziomie 95-100%). Takie przeciążenie może spowodować spadek wydajności, w wyniku czego kontrole są przetwarzane wolniej, niż powinny. W celu uniknięcia problemów wynikających z przeciążenia pamięci podręcznej historii zaimplementowano mechanizm ochrony przed takim scenariuszem.

Gdy pamięć podręczna historii serwera Zabbix jest pełna, dostęp do zapisu w pamięci podręcznej historii jest ograniczany, co wstrzymuje procesy zbierania danych przez serwer. Najczęstszym przypadkiem przeciążenia pamięci podręcznej historii jest sytuacja po przestoju serwera, gdy proxy przesyłają zebrane dane. Aby temu zapobiec, dodano ograniczanie przepustowości proxy (obecnie nie można go wyłączyć).

Gdy użycie pamięci podręcznej historii osiągnie 80%, serwer Zabbix przechodzi w tryb ograniczania przepustowości. W tym trybie serwer przyjmuje dane od proxy tylko wtedy, gdy użycie pamięci podręcznej historii spadnie poniżej 60%, rotując akceptowane proxy. Gdy użycie pamięci podręcznej historii spadnie poniżej 20%, serwer wraca do trybu normalnego.

Dodatkowo w trybie normalnym serwer Zabbix ogranicza pojedyncze proxy wysyłające bardzo duże pakiety (10 000+ rekordów), jeśli użycie pamięci podręcznej historii przekracza 60%. Ta decyzja jest podejmowana w momencie, gdy serwer ocenia przesyłanie danych przez proxy, dlatego nie zawsze może być od razu widoczna na wykresach użycia pamięci podręcznej historii (wewnętrzna pozycja `zabbix[wcache,history,pused]` i jej interwał aktualizacji mogą nie uwzględniać krótkich skoków).

Ten tryb ograniczania przepustowości będzie działał do momentu, gdy użycie pamięci podręcznej ponownie osiągnie 80%, spadnie do 20% albo lista ograniczania przepustowości będzie pusta. W pierwszym przypadku serwer ponownie przestanie przyjmować dane od proxy. W dwóch pozostałych przypadkach serwer zacznie działać normalnie, przyjmując dane od wszystkich proxy.

Powyższe informacje można przedstawić w następującej tabeli:

Użycie pamięci podręcznej zapisu historii	Tryb serwera Zabbix	Działanie serwera Zabbix
Osiąga 80%	Oczekiwanie	Przestaje przyjmować dane od proxy, ale utrzymuje *listę ograniczania przepustowości* (priorytetową listę proxy, z którymi należy skontaktować się później).
Osiąga 60%	Normalny, ale przygotowany do ograniczenia	Może odrzucać bardzo duże przesyłania z proxy (ponad 10 tys. rekordów) podczas podejmowania decyzji o przyjęciu danych; nadal przyjmuje inne dane od proxy.
Spada do 20%	Normalny	Usuwa listę ograniczania przepustowości i zaczyna normalnie przyjmować dane od proxy.

Do skorelowania tego zachowania serwera Zabbix z metryką można użyć wewnętrznej pozycji zabbix[wcache,history,pused].

Konfiguracja

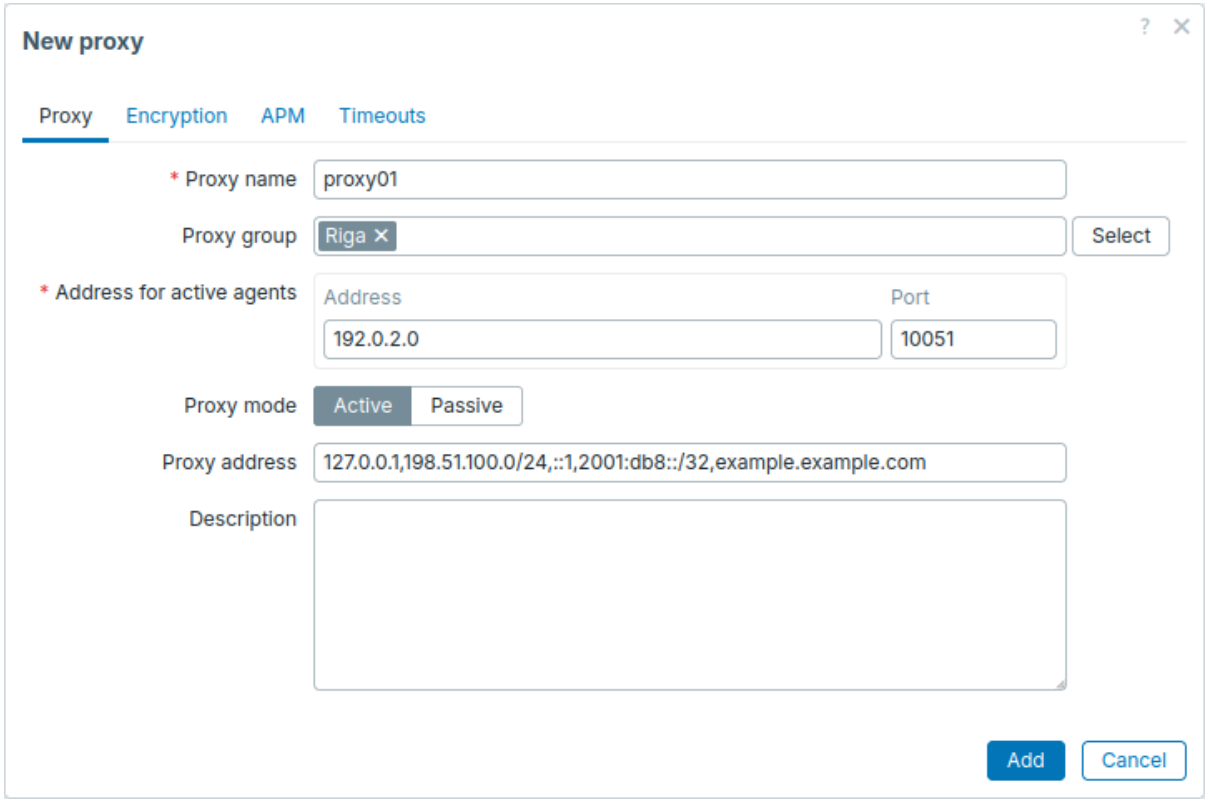
Po zainstalowaniu i skonfigurowaniu proxy nadszedł czas, aby skonfigurować je w frontendzie Zabbixa.

Dodawanie proxy

Note:
Zamiast instalować proxy we własnej infrastrukturze, możesz wdrożyć je w Zabbix Cloud. W tym celu kliknij *Create proxy in Cloud* w sekcji *Administration* > *Proxies*; Zabbix Cloud otworzy się w nowej karcie przeglądarki. Po utworzeniu tam węzła proxy skonfiguruj proxy w frontend Zabbix zgodnie z poniższym opisem. Więcej informacji znajdziesz w sekcji Konfigurowanie proxy Zabbix Cloud.

Aby skonfigurować proxy w frontend Zabbix:

1. Przejdź do sekcji *Administration* > *Proxies*.
2. Kliknij *Create proxy*.
3. Wprowadź parametry proxy w formularzu.

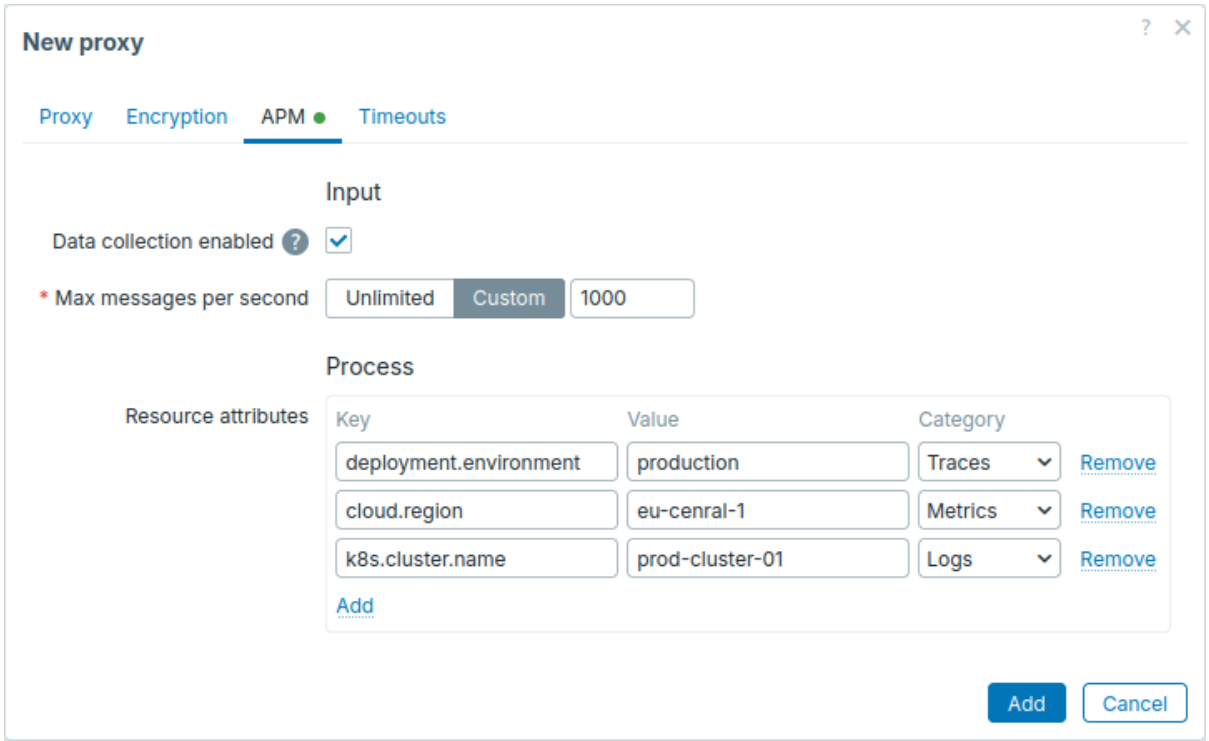


Parameter	Description
Proxy name	Wprowadź nazwę proxy. Musi być taka sama jak nazwa w parametrze *Hostname* w pliku konfiguracyjnym proxy.
Proxy group	Wybierz jedną grupę proxy na potrzeby równoważenia obciążenia/wysokiej dostępności proxy.
Address for active agents	Wprowadź adres, z którym muszą łączyć się monitorowane aktywne agenty lub sendery. Obsługiwane **tylko** przez agenty Zabbix 7.0 lub nowsze. Ten adres jest używany do łączenia zarówno z aktywnymi, jak i pasywnymi proxy. To pole jest dostępne tylko wtedy, gdy w polu *Proxy group* wybrano grupę proxy.
Address	Adres IP/nazwa DNS, z którym należy się połączyć.
Port	Numer portu TCP (domyślnie 10051), z którym należy się połączyć. Obsługiwane są makra użytkownika.
Proxy mode	Wybierz tryb proxy. **Active** - proxy połączy się z serwerem Zabbix i zażąda danych konfiguracyjnych **Passive** - serwer Zabbix łączy się z proxy *Uwaga*: bez szyfrowanej komunikacji (wrażliwe) dane konfiguracyjne proxy mogą stać się dostępne dla stron mających dostęp do portu trappera serwera Zabbix w przypadku używania aktywnego proxy. Jest to możliwe, ponieważ każdy może podszyć się pod aktywne proxy i zażądać danych konfiguracyjnych, jeśli uwierzytelnianie nie odbywa się lub adresy proxy nie są ograniczone w polu *Proxy address*.
Proxy address	Jeśli określono, żądania aktywnego proxy są akceptowane wyłącznie z tej listy rozdzielanych przecinkami adresów IP, opcjonalnie w notacji CIDR, lub nazw DNS aktywnego proxy Zabbix. To pole jest dostępne tylko wtedy, gdy w polu *Proxy mode* wybrano aktywne proxy. Makra nie są obsługiwane.
Interface	Wprowadź szczegóły interfejsu pasywnego proxy. To pole jest dostępne tylko wtedy, gdy w polu *Proxy mode* wybrano pasywne proxy.
Address	Adres IP/nazwa DNS pasywnego proxy.
Port	Numer portu TCP pasywnego proxy (domyślnie 10051). Obsługiwane są makra użytkownika.
Description	Wprowadź opis proxy.

Karta **Encryption** umożliwia wymaganie szyfrowanych połączeń z proxy.

Parameter	Description
Connections to proxy	Sposób, w jaki serwer łączy się z pasywnym proxy: bez szyfrowania (domyślnie), przy użyciu PSK (klucza wstępnie współdzielonego) lub certyfikatu.
Connections from proxy	Wybierz typy połączeń dozwolonych z aktywnego proxy. Można jednocześnie wybrać kilka typów połączeń (co jest przydatne podczas testowania i przełączania na inny typ połączenia). Domyślnie jest to „No encryption”.
Issuer	Dozwolony wystawca certyfikatu. Certyfikat jest najpierw weryfikowany przez CA (urząd certyfikacji). Jeśli jest prawidłowy i podpisany przez CA, pole *Issuer* może służyć do dalszego ograniczenia dozwolonych CA. To pole jest opcjonalne i przeznaczone do użycia, jeśli instalacja Zabbix korzysta z certyfikatów pochodzących z wielu CA.
Subject	Dozwolony podmiot certyfikatu. Certyfikat jest najpierw weryfikowany przez CA. Jeśli jest prawidłowy i podpisany przez CA, pole *Subject* może służyć do zezwolenia tylko na jedną wartość ciągu *Subject*. Jeśli to pole jest puste, akceptowany jest dowolny prawidłowy certyfikat podpisany przez skonfigurowane CA.
PSK identity	Ciąg identyfikatora klucza wstępnie współdzielonego. Nie umieszczaj poufnych informacji w identyfikatorze PSK, ponieważ jest on przesyłany przez sieć w postaci niezaszyfrowanej, aby poinformować odbiorcę, którego PSK należy użyć.
PSK	Klucz wstępnie współdzielony (ciąg szesnastkowy). Maksymalna długość: 512 cyfr szesnastkowych (klucz PSK o długości 256 bajtów), jeśli Zabbix używa biblioteki GnuTLS lub OpenSSL, albo 64 cyfry szesnastkowe (klucz PSK o długości 32 bajtów), jeśli Zabbix używa biblioteki mbed TLS (PolarSSL). Przykład: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

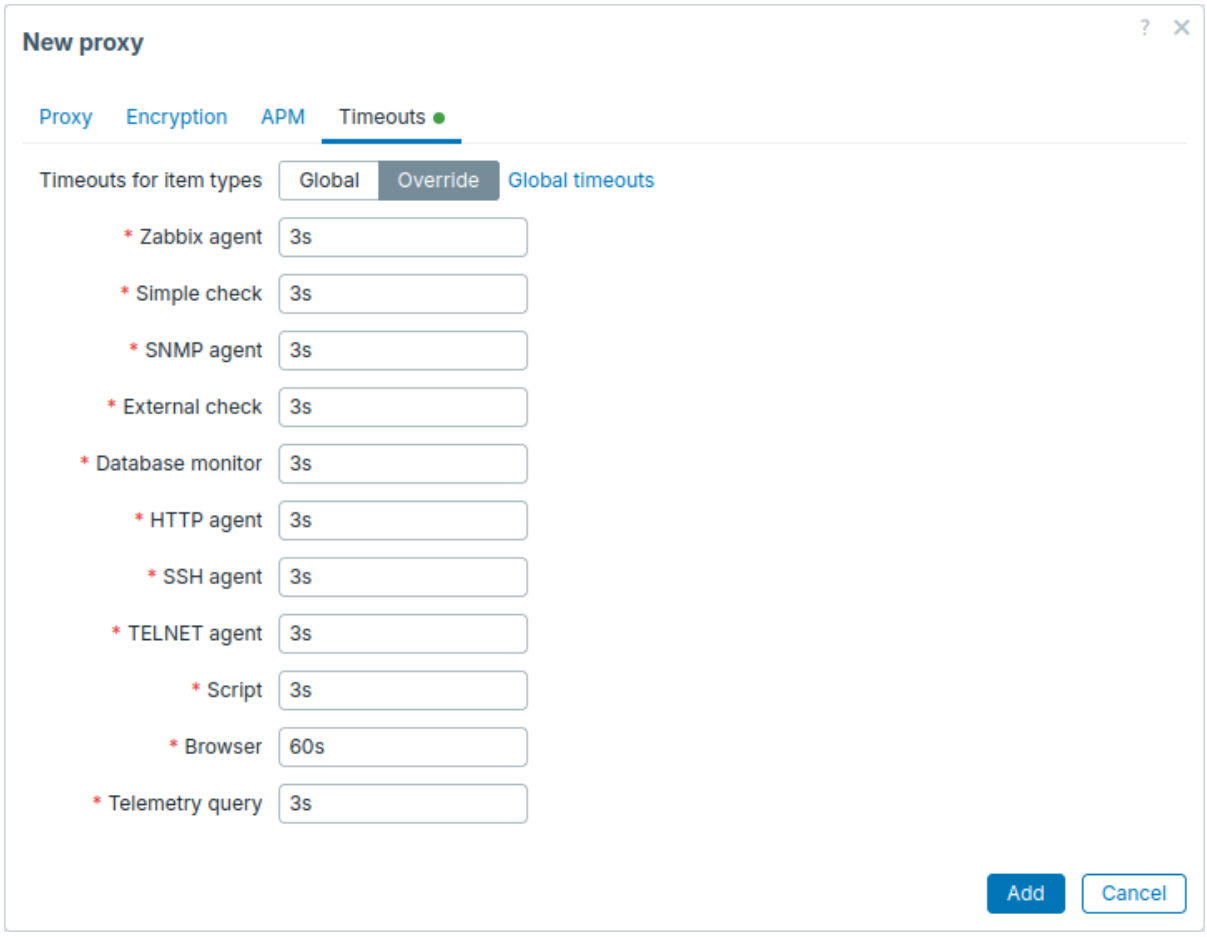
Karta **APM** umożliwia skonfigurowanie monitorowania wydajności aplikacji (APM) na proxy. Proxy może zbierać dane telemetryczne (ślady, metryki i logi) wysyłane przez aplikacje lub kolektory OpenTelemetry za pośrednictwem protokołu OpenTelemetry (OTLP/gRPC) i zapisywać te dane w źródle danych telemetrycznych.



Parameter	Description
Data collection enabled	Zaznacz pole wyboru, aby włączyć zbieranie danych telemetrycznych na proxy. Przed włączeniem zbierania danych telemetrycznych ustaw parametr `StartAPMCollector` na 1 w pliku konfiguracyjnym proxy (wymaga proxy Zabbix z obsługą APM).

Parameter	Description
Max messages per second	Ustaw maksymalną liczbę wiadomości na sekundę akceptowanych przez proxy: **Unlimited** - akceptuj wszystkie przychodzące wiadomości. **Custom** - akceptuj do określonej liczby wiadomości na sekundę. Wiadomość to żądanie OTLP wysłane przez aplikację, kolektor OpenTelemetry lub inne źródło danych telemetrycznych. Jej rozmiar zależy od nadawcy; na przykład kolektor OpenTelemetry może grupować tysiące spanów w jednej wiadomości, podczas gdy prosta aplikacja może wysyłać mniejsze wiadomości (np. jedną na ślad). Jeśli limit zostanie przekroczony, proxy odpowie kodem stanu gRPC `UNAVAILABLE` (błąd tymczasowy), więc nadawca może ponowić próbę wysłania wiadomości później, zależnie od swojej konfiguracji.
Add resource attributes	Określ atrybuty zasobów, które mają zostać dodane do przychodzących danych telemetrycznych wybranego typu (np. aby oznaczyć wszystkie dane odebrane przez to proxy jego środowiskiem lub regionem). Następnie możesz używać tych atrybutów do filtrowania danych telemetrycznych, na przykład w warunkach pozycji telemetry query. Jeśli przychodzące dane zawierają już atrybut zasobu o tej samej nazwie, zachowany zostanie atrybut przychodzący zamiast określonego tutaj.
Name	Wprowadź nazwę atrybutu (np. `deployment.environment`, `cloud.region`). Musi być unikatowa dla wybranego *Type*.
Value	Wprowadź wartość atrybutu (np. `production`, `eu-central-1`). Może być pusta.
Type	Wybierz typ przychodzących danych, do których ma zastosowanie atrybut: *Traces*, *Metrics* lub *Logs*.

Karta **Timeouts** umożliwia zastąpienie globalnych limitów czasu dla typów pozycji, które to obsługują.



Parameter	Description
Timeouts for item types	Ustaw limit czasu pozycji (na podstawie jej typu): **Global** - użyj globalnego limitu czasu (wyświetlanego w wyszarzonym polu *Timeout* dla każdego typu pozycji); **Override** - ustaw niestandardowy limit czasu (w polu *Timeout* dla każdego typu pozycji). Dozwolony zakres: 1 - 600 s (domyślnie: odziedziczony z globalnych limitów czasu). Obsługiwane są przyrostki czasu, np. 30s i 1m, oraz makra użytkownika. Kliknięcie łącza *Global timeouts* umożliwia skonfigurowanie globalnych limitów czasu. Pamiętaj, że łącze *Global timeouts* jest widoczne tylko dla użytkowników typu *Super admin* z uprawnieniami do sekcji frontend *Administration* > *General*. Pamiętaj, że chociaż limity czasu na poziomie proxy zastępują limity globalne, zostaną zastąpione przez indywidualne limity czasu pozycji, jeśli zostaną skonfigurowane.

Note:

Jeśli główna wersja proxy nie odpowiada głównej wersji serwera, ikona zostanie wyświetlona obok *Timeouts for item types*, a po najechaniu kursorem pojawi się komunikat „Timeouts disabled because the proxy and server versions do not match". W takich przypadkach proxy użyje parametru `Timeout` z pliku konfiguracyjnego proxy.

Formularz edycji istniejącego proxy zawiera następujące dodatkowe przyciski:

- *Refresh configuration* - odśwież konfigurację proxy;
- *Clone* - utwórz nowe proxy na podstawie właściwości istniejącego proxy;
- *Delete* - usuń proxy.

Konfiguracja hosta

Możesz określić, że pojedynczy host ma być monitorowany przez proxy lub grupę proxy, w formularzu konfiguracji hosta, używając pola *Monitored by*.



Masowa aktualizacja hosta to inny sposób określenia, że hosty powinny być monitorowane przez proxy lub grupę proxy.

Równoważenie obciążenia proxy i wysoka dostępność

Przegląd

proxy Zabbix można organizować w grupy proxy, aby umożliwić równoważenie obciążenia proxy i wysoką dostępność.

Równoważenie obciążenia proxy i wysoka dostępność oznaczają automatyczną redystrybucję hostów między proxy w ramach grupy proxy:

- Jeśli proxy przejdzie w tryb offline, jego hosty zostaną przeniesione do innych proxy, co pozwoli utrzymać wysoką dostępność proxy.
- Jeśli proxy ma znacznie większą/mniejszą liczbę hostów niż inne proxy, jego hosty zostaną przeniesione do innych proxy w celu zrównoważenia obciążenia proxy.

Redystrybucja hostów działa tylko między proxy w grupie, które spełniają następujące warunki:

- Proxy działają w wersji Zabbix 7.0 lub nowszej.
- Wersja proxy jest zgodna z wersją serwera Zabbix. Jeśli używany jest agent Zabbix (pasywny), wersja proxy musi być zgodna z wersją agenta. Aktywne agenty wymagają jedynie wersji Zabbix 7.0 lub nowszej.
- Grupa proxy ma stan online.
- Hosty są skonfigurowane tak, aby były monitorowane przez grupę proxy, a nie przez poszczególne proxy.

Note:
Stan grupy proxy można monitorować za pomocą wewnętrznych kontroli z użyciem dowolnego hosta przypisanego do grupy proxy. Aby jednak monitorować stan pojedynczego proxy w grupie, przypisz hosta do tego proxy; w przeciwnym razie wyniki mogą być niespójne.

Redystrybucja hostów

Równoważenie obciążenia proxy i wysoka dostępność są zarządzane przez serwer Zabbix za pośrednictwem menedżera grup proxy, który stale monitoruje stan wszystkich proxy w każdej grupie proxy oraz ich rozkład hostów.

Wysoka dostępność proxy w obrębie grupy jest zapewniana przez przełączanie awaryjne proxy: gdy proxy przechodzi w tryb offline, jego hosty są natychmiast redystrybuowane do innych proxy. Występuje również równoważenie obciążenia proxy, ponieważ hosty są przypisywane ponownie do proxy z najmniejszą liczbą przypisanych hostów.

Dodatkowo równoważenie obciążenia proxy jest uruchamiane, gdy liczba hostów proxy różni się od średniej dla grupy o co najmniej 10 hostów oraz o współczynnik 2 (nadmiar hostów lub niedobór hostów). Jeśli brak równowagi utrzymuje się po okresie karencji (10 x opóźnienie przełączania awaryjnego), grupa proxy zostaje umieszczona w kolejce do redystrybucji hostów.

Menedżer grup proxy redystrybuuje hosty według następującej logiki:

1. Oblicz średnią liczbę hostów na proxy.
2. Dla proxy z nadmiarem hostów — przenieś nadmiarowe hosty do puli nieprzypisanych hostów proxy.
3. Dla proxy z niedoborem hostów — oblicz, ile hostów jest potrzebnych do osiągnięcia równowagi.
4. Usuń wymaganą liczbę hostów z proxy z największą liczbą hostów.
5. Przenieś nieprzypisane hosty do proxy z najmniejszą liczbą hostów.

Przykłady redystrybucji hostów:

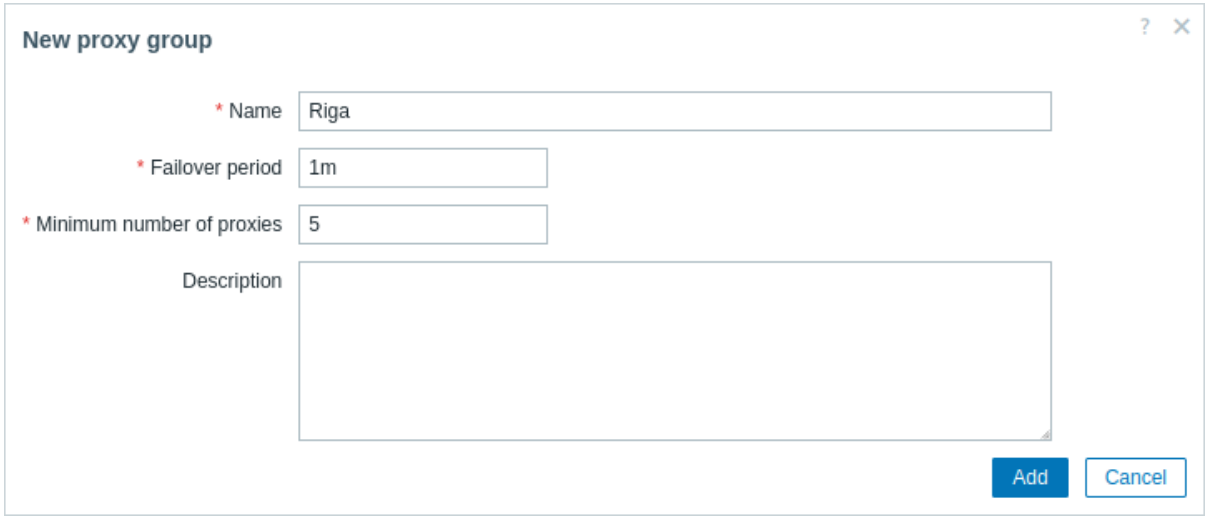
Hosty na proxy	Średnia dla grupy	Ponowne przypisanie hostów
100	50	Tak
60	50	Nie
40	50	Nie
25	50	Tak
15	5	Tak
10	5	Nie

Attention:
Posiadanie mniej niż 10 hostów monitorowanych przez grupę proxy może prowadzić do nierównomiernego rozkładu hostów między proxy w grupie.

Konfigurowanie grupy proxy

Aby skonfigurować grupę proxy w frontend Zabbix:

1. Przejdź do *Administracja* > *Grupy proxy*.
2. Kliknij *Utwórz grupę proxy*.
3. Wprowadź parametry grupy proxy w formularzu.



Parametr	Opis
Nazwa	Nazwa grupy proxy.
Okres przełączania awaryjnego	Okres w sekundach, w którym proxy w grupie proxy musi komunikować się z serwerem Zabbix, aby zostać uznane za dostępne (domyślnie: 1m; zakres: 10s-15m). Jeśli proxy nie skomunikuje się w tym okresie, jego stan zostanie zmieniony na *Niedostępne*, a jego hosty zostaną natychmiast rozdzielone między inne proxy. Równoważenie obciążenia proxy rozpoczyna się po upływie 10 x tego okresu. Obsługuje przyrostki czasu (np. 30s, 1m) oraz makra użytkownika.
Minimalna liczba proxy	Minimalna liczba dostępnych proxy wymagana do utrzymania dostępności grupy proxy (domyślnie: 1; zakres: 1-1000). Obsługuje makra użytkownika. Ta wartość powinna być mniejsza niż łączna liczba proxy w grupie. Na przykład w grupie składającej się z 10 proxy ustawienie minimalnej liczby na 10 spowoduje przejście grupy w stan niedostępności w przypadku awarii dowolnego proxy. Należy pamiętać, że dostępne proxy w niedostępnej grupie nadal działają normalnie, ale równoważenie obciążenia/wysoka dostępność nie będą realizowane.
Opis	Opis grupy proxy.
Proxy	Podczas edytowania grupy zawierającej proxy wyświetla listę maksymalnie pięciu proxy (jako linki lub zwykły tekst, w zależności od uprawnień użytkownika do proxy).

Konfigurowanie równoważenia obciążenia proxy

Aby korzystać z równoważenia obciążenia proxy, należy skonfigurować grupę proxy w frontend Zabbix (zobacz powyżej) i upewnić się, że hosty są monitorowane przez grupę proxy, a nie przez poszczególne proxy (można użyć funkcji masowej aktualizacji hostów, aby przenieść hosty z proxy do grupy proxy).

Jeśli używasz Zabbix agent, skonfiguruj go również w następujący sposób:

- W przypadku pasywnych kontroli wymień wszystkie proxy należące do grupy proxy w parametrze Server.
- W przypadku aktywnych kontroli zaleca się wymienienie wszystkich proxy należących do grupy proxy lub serwera Zabbix w parametrze ServerActive. Należy pamiętać, że z grupami proxy w trybie aktywnym będzie działać tylko Zabbix agent 7.0 (lub nowszy).

Jeśli parametr `ServerActive` zawiera tylko jedno proxy z grupy proxy (lub serwer Zabbix), agent nadal będzie mógł połączyć się z właściwym proxy. Po uruchomieniu usługi agenta i połączeniu się z określonym proxy agent otrzyma i zapisze w pamięci podręcznej pełną listę adresów IP proxy oraz ich bieżące obciążenie w grupie. Następnie aktywne kontrole zostaną przekierowane do właściwego dostępnego proxy dla hosta na podstawie bieżącego przypisania proxy do hosta w grupie proxy.

Warning:
Określenie tylko jednego proxy w parametrze `ServerActive` Zabbix agent może prowadzić do utraty danych monitorowania, jeśli agent zostanie uruchomiony lub ponownie uruchomiony, gdy określone proxy będzie niedostępne.

Note:
Podczas korzystania z Zabbix sender żądania danych są również przekierowywane do właściwego dostępnego proxy dla hosta na podstawie bieżącego przypisania proxy do hosta w grupie proxy. Jeśli jednak wysyłasz wartości wielu hostów z pliku wejściowego, użyj opcji *−g*, aby zapobiec wysyłaniu danych do niewłaściwego proxy.

Zabbix agent musi również mieć możliwość łączenia się przez zaporę sieciową ze wszystkimi proxy w grupie proxy. W przeciwnym razie aktywne kontrole mogą zawieszać się lub kończyć niepowodzeniem podczas przekierowania albo przełączania awaryjnego. Na przykład:

- Podczas aktywnych kontroli proxy może przekierować agenta do innego proxy. Jeśli to proxy jest blokowane przez zaporę sieciową, komunikacja zawiesi się podczas oczekiwania na odpowiedź.
- W stabilnych konfiguracjach wysokiej dostępności bez niedawnego ponownego równoważenia agenty mogą nigdy nie kontaktować się z zapasowymi proxy. Jeśli reguły zapory sieciowej uległy zmianie i nie zostały przetestowane, przełączanie awaryjne może się nie powieść.

Testowanie równoważenia obciążenia proxy

Aby przetestować równoważenie obciążenia proxy:

1. Skonfiguruj grupę proxy.
2. Upewnij się, że grupa proxy ma stan online.

3. Upewnij się, że hosty są monitorowane przez grupę proxy, a nie przez poszczególne proxy (możesz użyć funkcji masowej aktualizacji hostów, aby przenieść hosty z proxy do grupy proxy).
4. Odczekaj kilka sekund na aktualizację konfiguracji i rozdzielenie hostów między proxy w grupie proxy. Obserwuj zmianę, odświeżając listę hostów w sekcji *Administracja* > *Proxy*.

Ważne uwagi

- Pułapki SNMP nie są obsługiwane przez proxy w grupie proxy.
- Sprawdzenia zależne od konfiguracji zewnętrznej (np. skrypty dla sprawdzeń zewnętrznych lub konfiguracja ODBC dla sprawdzeń baz danych) muszą mieć taką samą konfigurację na wszystkich proxy w grupie proxy.
- Sprawdzenia baz danych wymagają rozszerzonych uprawnień do obiektu/serwera bazy danych.
- Hosty VMware monitorowane przez grupę proxy są losowo rozdzielane między proxy w grupie. Powoduje to, że każde proxy buforuje wszystkie dane VMware, co skutkuje dodatkowym obciążeniem vCenter.
- Hosty utworzone na podstawie danych automatycznej rejestracji z proxy w grupie proxy są ustawiane jako monitorowane przez tę grupę proxy. Jednak hosty utworzone na podstawie danych wykrywania sieci z proxy w grupie proxy są ustawiane jako monitorowane przez to proxy.

Synchronizacja konfiguracji monitorowania

Przegląd

Ta strona zawiera szczegóły dotyczące aktualizacji konfiguracji monitorowania dla proxy, tj. sposobu synchronizacji zmian wprowadzonych w konfiguracji monitorowania na serwerze z proxy.

Aktualizacja przyrostowa

Aktualizacja konfiguracji proxy jest przyrostowa. Podczas synchronizacji konfiguracji aktualizowane są tylko zmodyfikowane encje (dlatego jeśli żadna encja nie została zmodyfikowana, nic nie zostanie wysłane). Takie podejście pozwala oszczędzać zasoby i ustawić krótszy interwał (niemal natychmiastowy) aktualizacji konfiguracji proxy.

Zmiany konfiguracji proxy są śledzone za pomocą numerów rewizji. Tylko encje z rewizjami większymi niż rewizja konfiguracji proxy są uwzględniane w danych konfiguracyjnych wysyłanych do proxy.

Encje uwzględniane podczas synchronizacji konfiguracji są następujące:

Encja	Szczegóły
dane TLS automatycznej rejestracji	Wszystkie dane TLS automatycznej rejestracji.
wyrażenia	Wszystkie wyrażenia (wyrażenia regularne, tabele wyrażeń).
konfiguracja globalna	Konfiguracja globalna zdefiniowana w tabeli 'config'
host	Wszystkie właściwości, interfejsy, dane inwentaryzacyjne, pozycje, przetwarzanie wstępne pozycji, parametry pozycji i scenariusze web hosta.
makra hosta	Wszystkie makra zdefiniowane na hoście oraz wszystkie identyfikatory szablonów z nim powiązane.
reguła wykrywania proxy	Reguły wykrywania i kontrole przypisane do proxy.

Oznacza to, że:

- Jeśli pozycja zostanie zmieniona na **hoście**, cała konfiguracja tego hosta zostanie zsynchronizowana.
- Jeśli zostanie zmienione **wyrażenie regularne**, wszystkie wyrażenia regularne zostaną zsynchronizowane.

Wyjątkiem są makra hosta, które są wysyłane również wtedy, gdy na hoście została zmieniona dowolna inna konfiguracja.

Polecenie `-R config_cache_reload` na proxy również zainicjuje aktualizację przyrostową.

Należy pamiętać, że pełna synchronizacja konfiguracji zostanie przeprowadzona podczas uruchamiania/ponownego uruchamiania proxy, przełączenia HA, zmiany tokenu sesji lub niepowodzenia aktualizacji konfiguracji na proxy, na przykład gdy połączenie zostało przerwane podczas odbierania danych konfiguracyjnych.

Parametry konfiguracji

Parametr **ProxyConfigFrequency** określa, jak często konfiguracja proxy jest synchronizowana z serwerem (domyślnie co 10 sekund).

Należy pamiętać, że ProxyConfigFrequency jest:

- parametrem serwera dla proxy pasywnych

- parametrem proxy dla proxy aktywnych

W przypadku proxy aktywnych ProxyConfigFrequency jest nowym parametrem od Zabbix 6.4 i musi być używany zamiast przestarzałego już ConfigFrequency.

Attention:
Jeśli używane są zarówno ProxyConfigFrequency, jak i ConfigFrequency, proxy zapisze błąd w logu i zakończy działanie.

Obliczanie kolejek podczas konserwacji

Attention:
Proxy Zabbix nie jest świadomy okresów konserwacji; szczegóły można znaleźć w sekcji Obliczanie kolejek podczas konserwacji.

Protokół wymiany danych serwer-proxy

Serwer - wymiana danych z proxy opiera się na formacie JSON.

Wiadomości żądania i odpowiedzi muszą zaczynać się od nagłówka i długości danych.

Proxy pasywny

Żądanie konfiguracji

Serwer najpierw wyśle puste żądanie `proxy config`. To żądanie jest wysyłane co `ProxyConfigFrequency` (parametr konfiguracji serwera) sekund.

proxy odpowiada bieżącą wersją proxy, tokenem sesji i rewizją konfiguracji. Serwer odpowiada danymi konfiguracji, które należy zaktualizować.

name	value type	description
server→proxy:		
request	*string*	„proxy config”
proxy→server:		
version	*string*	Wersja proxy (<major>.<minor>.<build>).
session	*string*	Token sesji konfiguracji proxy.
config_revision	*number*	Rewizja konfiguracji proxy.
server→proxy:		
full_sync	*number*	1 - jeśli wysyłane są pełne dane konfiguracji; w przeciwnym razie brak (opcjonalne).
data	*array*	Obiekt danych tabel. Brak, jeśli konfiguracja nie została zmieniona (opcjonalne).
<table>	*object*	Jeden lub więcej obiektów z danymi <table> (opcjonalne, zależnie od zmian).
fields	*array*	Tablica nazw pól.
-	*string*	Nazwa pola.
data	*array*	Tablica wierszy.
-	*array*	Tablica kolumn.
-	*string*,*number*	Wartość kolumny, której typ zależy od typu kolumny w schemacie bazy danych.
macro.secrets	*object*	Informacje o makrach poufnych; brak, jeśli nie ma zmian w makrach przechowywanych w magazynie (opcjonalne).
config_revision	*number*	Rewizja pamięci podręcznej konfiguracji - wysyłana wraz z danymi konfiguracji (opcjonalne).
del_hostids	*array*	Tablica usuniętych identyfikatorów hostów (opcjonalne).
-	*number*	Identyfikator hosta.
del_macro_hostids	*array*	Tablica identyfikatorów hostów, dla których usunięto wszystkie makra (opcjonalne).
-	*number*	Identyfikator hosta.
proxy→server:		

name	value type	description
response	*string*	Informacja o powodzeniu żądania („success” lub „failed”).
version	*string*	Wersja proxy (<major>.<minor>.<build>).

Przykład:

serwer→proxy:

```
{
  "request":"proxy config"
}
```

proxy→serwer:

```
{
  "version": "8.0.0",
  "session": "0033124949800811e5686dbfd9bcea98",
  "config_revision": 0
}
```

serwer→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, "Local network", "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, "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
        "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→serwer:


```
{
  "response": "success",
  "version": "8.0.0"
}
```


Żądanie danych proxy

Żądanie `proxy data` służy do pobierania dostępności interfejsów hosta, danych historycznych, wykrywania oraz autorejestracji z

proxy. To żądanie jest wysyłane co `ProxyDataFrequency` (parametr konfiguracji serwera) sekund.

name	value type	description
server→proxy:		
request	*string*	'proxy data'
proxy→server:		
session	*string*	Token sesji danych.
interface availability	*array*	*(optional)* Tablica obiektów danych dostępności interfejsów.
interfaceid	*number*	Identyfikator interfejsu.
available	*number*	Dostępność interfejsu: **0**, *INTERFACE_AVAILABLE_UNKNOWN* - nieznana **1**, *INTERFACE_AVAILABLE_TRUE* - dostępny **2**, *INTERFACE_AVAILABLE_FALSE* - niedostępny
error	*string*	Komunikat błędu interfejsu lub pusty ciąg znaków.
history data	*array*	*(optional)* Tablica obiektów danych historycznych.
itemid	*number*	Identyfikator pozycji.
clock	*number*	Znacznik czasu wartości pozycji (sekundy).
ns	*number*	Znacznik czasu wartości pozycji (nanosekundy).
value	*string*	*(optional)* Wartość pozycji.
id	*number*	Identyfikator wartości (rosnący licznik, unikalny w ramach jednej sesji danych).
timestamp	*number*	*(optional)* Znacznik czasu dla pozycji typu log.
source	*string*	*(optional)* Wartość źródła pozycji eventlog.
severity	*number*	*(optional)* Wartość ważności pozycji eventlog.
eventid	*number*	*(optional)* Wartość eventid pozycji eventlog.
state	*string*	*(optional)* Stan pozycji: **0**, *ITEM_STATE_NORMAL* **1**, *ITEM_STATE_NOTSUPPORTED*
lastlogsize	*number*	*(optional)* Ostatni rozmiar logu dla pozycji typu log.
mtime	*number*	*(optional)* Czas modyfikacji pozycji typu log.
discovery data	*array*	*(optional)* Tablica obiektów danych wykrywania.
clock	*number*	Znacznik czasu danych wykrywania.
druleid	*number*	Identyfikator reguły wykrywania.
dcheckid	*number*	Identyfikator sprawdzenia wykrywania lub null dla danych reguły wykrywania.
type	*number*	Typ sprawdzenia wykrywania: **-1** dane reguły wykrywania **0**, *SVC_SSH* - sprawdzenie usługi SSH **1**, *SVC_LDAP* - sprawdzenie usługi LDAP **2**, *SVC_SMTP* - sprawdzenie usługi SMTP **3**, *SVC_FTP* - sprawdzenie usługi FTP **4**, *SVC_HTTP* - sprawdzenie usługi HTTP **5**, *SVC_POP* - sprawdzenie usługi POP **6**, *SVC_NNTP* - sprawdzenie usługi NNTP **7**, *SVC_IMAP* - sprawdzenie usługi IMAP **8**, *SVC_TCP* - sprawdzenie dostępności portu TCP **9**, *SVC_AGENT* - agent Zabbixa **10**, *SVC_SNMPv1* - agent SNMPv1 **11**, *SVC_SNMPv2* - agent SNMPv2 **12**, *SVC_ICMPPING* - ping ICMP **13**, *SVC_SNMPv3* - agent SNMPv3 **14**, *SVC_HTTPS* - sprawdzenie usługi HTTPS **15**, *SVC_TELNET* - sprawdzenie dostępności Telnet
ip	*string*	Adres IP hosta.
dns	*string*	Nazwa DNS hosta.

name	value type	description
port	*number*	*(optional)* Numer portu usługi.
key_	*string*	*(optional)* Klucz pozycji dla sprawdzenia wykrywania typu **9** *SVC_AGENT*
value	*string*	*(optional)* Wartość odebrana z usługi, dla większości usług może być pusta.
status	*number*	*(optional)* Stan usługi: **0**, *DOBJECT_STATUS_UP* - usługa działa **1**, *DOBJECT_STATUS_DOWN* - usługa nie działa
auto reg- is- tra- tion	*array*	*(optional)* Tablica obiektów danych autorejestracji.
clock	*number*	Znacznik czasu danych autorejestracji.
host	*string*	Nazwa hosta.
ip	*string*	*(optional)* Adres IP hosta.
dns	*string*	*(optional)* Rozwiązana nazwa DNS z adresu IP.
port	*string*	*(optional)* Port hosta.
host_metadata	*string*	*(optional)* Metadane hosta wysłane przez agent (na podstawie parametru konfiguracji agenta HostMetadata lub HostMetadataItem).
tasks	*array*	*(optional)* Tablica zadań.
type	*number*	Typ zadania: **0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - wynik zdalnego polecenia
status	*number*	Stan wykonania zdalnego polecenia: **0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - zdalne polecenie wykonane pomyślnie **1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - zdalne polecenie nie powiodło się
error	*string*	*(optional)* Komunikat błędu.
parent_taskid	*number*	Identyfikator nadrzędnego zadania.
more	*number*	*(optional)* 1 - do wysłania pozostały kolejne dane historyczne.
clock	*number*	*(optional)* Znacznik czasu przesyłania danych (sekundy).
ns	*number*	*(optional)* Znacznik czasu przesyłania danych (nanosekundy).
version	*string*	Wersja proxy (<major>.<minor>.<build>).
server→proxy:		
response	*string*	Informacja o powodzeniu żądania (’success’ lub ’failed’).
tasks	*array*	*(optional)* Tablica zadań.
type	*number*	Typ zadania: **1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - zdalne polecenie
clock	*number*	Czas utworzenia zadania.
ttl	*number*	Czas w sekundach, po którym zadanie wygasa.
commandtype	*number*	Typ zdalnego polecenia: **0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - użyj własnego skryptu **1**, *ZBX_SCRIPT_TYPE_IPMI* - użyj IPMI **2**, *ZBX_SCRIPT_TYPE_SSH* - użyj SSH **3**, *ZBX_SCRIPT_TYPE_TELNET* - użyj Telnet **4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - użyj skryptu globalnego (obecnie funkcjonalnie równoważny własnemu skryptowi)
command	*string*	Zdalne polecenie do wykonania.
execute_on	*number*	Cel wykonania dla własnych skryptów: **0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - wykonaj skrypt na agent **1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - wykonaj skrypt na serwer **2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - wykonaj skrypt na proxy
port	*number*	*(optional)* Port dla poleceń Telnet i SSH.
authtype	*number*	*(optional)* Typ uwierzytelniania dla poleceń SSH.
username	*string*	*(optional)* Nazwa użytkownika dla poleceń Telnet i SSH.
password	*string*	*(optional)* Hasło dla poleceń Telnet i SSH.

name	value type	description
publickey	*string*	*(optional)* Klucz publiczny dla poleceń SSH.
privatekey	*string*	*(optional)* Klucz prywatny dla poleceń SSH.
parent_taskid	*number*	Identyfikator nadrzędnego zadania.
hostid	*number*	Identyfikator hosta docelowego.

Przykład:

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
        }
    ]
}
```

Żądanie zadań

Wymiana `proxy tasks` obsługuje dostarczanie i potwierdzanie zadań poleceń zdalnych między serwerem a pasywnym proxy. Gdy serwer odpytuje proxy o wyniki zadań, wysyła puste żądanie `proxy tasks` w interwale zdefiniowanym przez `ProxyDataFrequency`. Proxy odpowiada bieżącą wersją oraz wszystkimi oczekującymi wynikami zadań (w tym znacznikami czasu). Następnie serwer potwierdza odbiór, zwracając `response`, i może dołączyć nowe zadania do wykonania przez proxy.

name	value type	description
server→proxy:		
request	*string*	'proxy tasks'
proxy→server:		
version	*string*	Wersja proxy (<major>.<minor>.<build>).
clock	*number*	*(opcjonalnie)* Znacznik czasu przesyłania danych (sekundy).
ns	*number*	*(opcjonalnie)* Znacznik czasu przesyłania danych (nanosekundy).
server→proxy:		
response	*string*	Informacja o powodzeniu żądania ('success' lub 'failed').

Przykład:

server→proxy:

```
{
  "request":"proxy tasks"
}
```

proxy→server:

```
{
  "version":"7.0.0",
  "clock":1721059872,
  "ns":848141522
}
```

server→proxy:

```
{
  "response":"success"
}
```

Aktywny proxy

Żądanie konfiguracji

Żądanie `proxy config` jest wysyłane przez aktywny proxy w celu pobrania danych konfiguracji proxy. To żądanie jest wysyłane co `ProxyConfigFrequency` (parametr konfiguracji proxy) sekund.

name	value type	description
proxy→server:		
request	*string*	'proxy config'
host	*string *	Nazwa proxy.
version	*string*	Wersja proxy (<major>.<minor>.<build>).
session	*string*	Token sesji konfiguracji proxy.
config_revision	*number*	Wersja konfiguracji proxy.
hostmap_revision	*number*	Bieżąca wersja mapowania hostów do proxy.
server→proxy:		
full_sync	*number*	1 - jeśli wysyłane są pełne dane konfiguracji; w przeciwnym razie brak (opcjonalne).
data	*array*	Obiekt danych tabel. Brak, jeśli konfiguracja nie została zmieniona (opcjonalne).
<table>	*object*	Jeden lub więcej obiektów z danymi <table> (opcjonalne, zależnie od zmian).
fields	*array*	Tablica nazw pól.
-	*string*	Nazwa pola.
data	*array*	Tablica wierszy.
-	*array*	Tablica kolumn.
-	*string,number*	Wartość kolumny, której typ zależy od typu kolumny w schemacie bazy danych.
macro.secrets	*object*	Informacje o makrach tajnych; brak, jeśli nie ma zmian w makrach przechowywanych w magazynie (opcjonalne).
proxy_group	*string*	Nazwa grupy proxy, do której należy proxy.
config_revision	*number*	Wersja pamięci podręcznej konfiguracji - wysyłana wraz z danymi konfiguracji (opcjonalne).
del_hostids	*array*	Tablica usuniętych identyfikatorów hostów (opcjonalne).
-	*number*	Identyfikator hosta.
del_macro_hostids	*array*	Tablica identyfikatorów hostów, dla których usunięto wszystkie makra (opcjonalne).
-	*number*	Identyfikator hosta.

Przykład:

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
}
```

Żądanie danych proxy

Żądanie `proxy data` jest wysyłane przez proxy w celu przekazania danych o dostępności interfejsów hosta, historii, wykrywania i autorejestracji. To żądanie jest wysyłane co `DataSenderFrequency` sekund (parametr konfiguracji proxy). Należy pamiętać, że aktywny proxy nadal będzie odpytywać serwer Zabbix co sekundę o zadania zdalnych poleceń (z pustym żądaniem `proxy data`).

name	value type	description
proxy→server:		
request	*string*	'proxy data'
host	*string*	Nazwa proxy.
session	*string*	Token sesji danych.
interface availability	*array*	*(opcjonalnie)* Tablica obiektów danych o dostępności interfejsów.
interfaceid	*number*	Identyfikator interfejsu.
available	*number*	Dostępność interfejsu: **0**, *INTERFACE_AVAILABLE_UNKNOWN* - nieznana **1**, *INTERFACE_AVAILABLE_TRUE* - dostępny **2**, *INTERFACE_AVAILABLE_FALSE* - niedostępny
error	*string*	Komunikat błędu interfejsu lub pusty ciąg znaków.
history data	*array*	*(opcjonalnie)* Tablica obiektów danych historycznych.
itemid	*number*	Identyfikator pozycji.
clock	*number*	Znacznik czasu wartości pozycji (sekundy).
ns	*number*	Znacznik czasu wartości pozycji (nanosekundy).
value	*string*	*(opcjonalnie)* Wartość pozycji.
id	*number*	Identyfikator wartości (rosnący licznik, unikalny w ramach jednej sesji danych).
timestamp	*number*	*(opcjonalnie)* Znacznik czasu dla pozycji typu log.
source	*string*	*(opcjonalnie)* Wartość źródła pozycji eventlog.
severity	*number*	*(opcjonalnie)* Wartość ważności pozycji eventlog.
eventid	*number*	*(opcjonalnie)* Wartość eventid pozycji eventlog.
state	*string*	*(opcjonalnie)* Stan pozycji: **0**, *ITEM_STATE_NORMAL* **1**, *ITEM_STATE_NOTSUPPORTED*
lastlogsize	*number*	*(opcjonalnie)* Ostatni rozmiar logu dla pozycji typu log.
mtime	*number*	*(opcjonalnie)* Czas modyfikacji dla pozycji typu log.
discovery data	*array*	*(opcjonalnie)* Tablica obiektów danych wykrywania.

name	value type	description
clock	*number*	Znacznik czasu danych wykrywania.
druleid	*number*	Identyfikator reguły wykrywania.
dcheckid	*number*	Identyfikator sprawdzenia wykrywania lub null dla danych reguły wykrywania.
type	*number*	Typ sprawdzenia wykrywania: **-1** dane reguły wykrywania **0**, *SVC_SSH* - sprawdzenie usługi SSH **1**, *SVC_LDAP* - sprawdzenie usługi LDAP **2**, *SVC_SMTP* - sprawdzenie usługi SMTP **3**, *SVC_FTP* - sprawdzenie usługi FTP **4**, *SVC_HTTP* - sprawdzenie usługi HTTP **5**, *SVC_POP* - sprawdzenie usługi POP **6**, *SVC_NNTP* - sprawdzenie usługi NNTP **7**, *SVC_IMAP* - sprawdzenie usługi IMAP **8**, *SVC_TCP* - sprawdzenie dostępności portu TCP **9**, *SVC_AGENT* - agent Zabbix **10**, *SVC_SNMPv1* - agent SNMPv1 **11**, *SVC_SNMPv2* - agent SNMPv2 **12**, *SVC_ICMPPING* - ping ICMP **13**, *SVC_SNMPv3* - agent SNMPv3 **14**, *SVC_HTTPS* - sprawdzenie usługi HTTPS **15**, *SVC_TELNET* - sprawdzenie dostępności Telnet
ip	*string*	Adres IP hosta.
dns	*string*	Nazwa DNS hosta.
port	*number*	*(opcjonalnie)* Numer portu usługi.
key_	*string*	*(opcjonalnie)* Klucz pozycji dla sprawdzenia wykrywania typu **9** *SVC_AGENT*
value	*string*	*(opcjonalnie)* Wartość odebrana z usługi, dla większości usług może być pusta.
status	*number*	*(opcjonalnie)* Stan usługi: **0**, *DOBJECT_STATUS_UP* - usługa działa **1**, *DOBJECT_STATUS_DOWN* - usługa nie działa
autoregistration	*array*	*(opcjonalnie)* Tablica obiektów danych autorejestracji.
clock	*number*	Znacznik czasu danych autorejestracji.
host	*string*	Nazwa hosta.
ip	*string*	*(opcjonalnie)* Adres IP hosta.
dns	*string*	*(opcjonalnie)* Rozwiązana nazwa DNS z adresu IP.
port	*string*	*(opcjonalnie)* Port hosta.
host_metadata	*string*	*(opcjonalnie)* Metadane hosta wysyłane przez agent (na podstawie parametru konfiguracji agenta HostMetadata lub HostMetadataItem).
tasks	*array*	*(opcjonalnie)* Tablica zadań.
type	*number*	Typ zadania: **0**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT* - wynik zdalnego polecenia
status	*number*	Stan wykonania zdalnego polecenia: **0**, *ZBX_TM_REMOTE_COMMAND_COMPLETED* - zdalne polecenie wykonane pomyślnie **1**, *ZBX_TM_REMOTE_COMMAND_FAILED* - zdalne polecenie nie powiodło się
error	*string*	*(opcjonalnie)* Komunikat błędu.
parent_taskid	*number*	ID zadania nadrzędnego.
more	*number*	*(opcjonalnie)* 1 - do wysłania pozostały kolejne dane historyczne
clock	*number*	*(opcjonalnie)* Znacznik czasu przesyłania danych (sekundy).
ns	*number*	*(opcjonalnie)* Znacznik czasu przesyłania danych (nanosekundy).
version	*string*	Wersja proxy (<major>.<minor>.<build>).
server→proxy:		
response	*string*	Informacja o powodzeniu żądania (’success’ lub ’failed’).

name	value type	description
upload	*string*	Kontrola przesyłania danych historycznych (historia, autorejestracja, dostępność hosta, wykrywanie sieci). Możliwe wartości: **enabled** - normalne działanie **disabled** - serwer nie przyjmuje danych (możliwe, że z powodu przekroczenia wewnętrznej pamięci podręcznej)
tasks	*array*	*(opcjonalnie)* Tablica zadań.
type	*number*	Typ zadania: **1**, *ZBX_TM_TASK_PROCESS_REMOTE_COMMAND* - zdalne polecenie
clock	*number*	Czas utworzenia zadania.
ttl	*number*	Czas w sekundach, po którym zadanie wygasa.
commandtype	*number*	Typ zdalnego polecenia: **0**, *ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT* - użyj niestandardowego skryptu **1**, *ZBX_SCRIPT_TYPE_IPMI* - użyj IPMI **2**, *ZBX_SCRIPT_TYPE_SSH* - użyj SSH **3**, *ZBX_SCRIPT_TYPE_TELNET* - użyj Telnet **4**, *ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT* - użyj globalnego skryptu (obecnie funkcjonalnie równoważny niestandardowemu skryptowi)
command	*string*	Zdalne polecenie do wykonania.
execute_on	*number*	Cel wykonania dla niestandardowych skryptów: **0**, *ZBX_SCRIPT_EXECUTE_ON_AGENT* - wykonaj skrypt na agent **1**, *ZBX_SCRIPT_EXECUTE_ON_SERVER* - wykonaj skrypt na serwer **2**, *ZBX_SCRIPT_EXECUTE_ON_PROXY* - wykonaj skrypt na proxy
port	*number*	*(opcjonalnie)* Port dla poleceń Telnet i SSH.
authtype	*number*	*(opcjonalnie)* Typ uwierzytelniania dla poleceń SSH.
username	*string*	*(opcjonalnie)* Nazwa użytkownika dla poleceń Telnet i SSH.
password	*string*	*(opcjonalnie)* Hasło dla poleceń Telnet i SSH.
publickey	*string*	*(opcjonalnie)* Klucz publiczny dla poleceń SSH.
privatekey	*string*	*(opcjonalnie)* Klucz prywatny dla poleceń SSH.
parent_taskid	*number*	ID zadania nadrzędnego.
hostid	*number*	ID hosta docelowego.

Przykład:

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
```

```
}]
}
```

Parametry konfiguracji: Zabbix proxy

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania proxy Zabbix.

Określ wartości parametrów w pliku konfiguracji proxy (`zabbix_proxy.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
AllowRoot	Zezwalaj na uruchamianie proxy jako `root`.
AllowUnsupportedDBVersions	Zezwalaj proxy na pracę z nieobsługiwanymi wersjami bazy danych.
APMListenIP	Określ adres IP, na którym kolektor APM powinien nasłuchiwać przychodzących danych telemetrycznych.
APMListenPort	Określ port, na którym kolektor APM powinien nasłuchiwać przychodzących danych telemetrycznych.
APMTLSCAFile	Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) służące do weryfikowania certyfikatów aplikacji lub kolektorów OpenTelemetry wysyłających dane do kolektora APM.
APMTLSCertFile	Określ pełną ścieżkę do pliku zawierającego certyfikat kolektora APM lub łańcuch certyfikatów.
APMTLSKeyFile	Określ pełną ścieżkę do pliku zawierającego klucz prywatny kolektora APM.
CacheSize	Ustaw rozmiar pamięci podręcznej konfiguracji.
ConfigFrequency	Ten parametr jest **przestarzały** (użyj ProxyConfigFrequency). Ustaw częstotliwość pobierania danych konfiguracji z serwera Zabbix.
DataSenderFrequency	Ustaw częstotliwość wysyłania zebranych danych do serwera Zabbix.
DBHost	Określ host bazy danych (lub katalog gniazda).
DBName	Określ nazwę bazy danych lub ścieżkę do pliku bazy danych SQLite3.
DBPassword	Określ hasło do bazy danych.
DBPort	Określ port hosta bazy danych Zabbix.
DBSchema	Określ nazwę schematu bazy danych. Używane w przypadku PostgreSQL.
DBSocket	Określ ścieżkę do pliku gniazda MySQL.
DBUser	Określ użytkownika bazy danych.
DBTLSConnect	Określ wartość wymuszającą połączenie TLS z bazą danych.
DBTLSCAFile	Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) służące do weryfikacji certyfikatu bazy danych.
DBTLSCertFile	Określ pełną ścieżkę do pliku zawierającego certyfikat proxy Zabbix używany do uwierzytelniania w bazie danych.
DBTLSKeyFile	Określ pełną ścieżkę do pliku zawierającego klucz prywatny używany do uwierzytelniania w bazie danych.
DBTLSCipher	Określ listę szyfrów szyfrowania dozwolonych przez proxy Zabbix dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko w przypadku MySQL.
DBTLSCipher13	Określ listę zestawów szyfrów dozwolonych przez proxy Zabbix dla protokołu TLS v1.3. Obsługiwane tylko w przypadku MySQL, począwszy od wersji 8.0.16.
DebugLevel	Ustaw poziom debugowania.
EnableRemoteCommands	Zezwalaj na zdalne polecenia z serwera Zabbix.
ExternalScripts	Określ lokalizację zewnętrznych skryptów.
Fping6Location	Określ lokalizację fping6.
FpingLocation	Określ lokalizację fping.
HistoryCacheSize	Ustaw rozmiar pamięci podręcznej historii.
HistoryIndexCacheSize	Ustaw rozmiar pamięci podręcznej indeksu historii.
Hostname	Określ nazwę proxy.
HostnameItem	Określ pozycję używaną do ustawiania wartości Hostname (jeśli nie została zdefiniowana).
HousekeepingFrequency	Ustaw częstotliwość procedury porządkowania (w godzinach).
Include	Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracji.
JavaGateway	Określ adres IP (lub nazwę hosta) Java gateway Zabbix.

Parametr	Opis
JavaGatewayPort	Określ port Java gateway Zabbix.
ListenBacklog	Ustaw maksymalną liczbę oczekujących połączeń w kolejce TCP.
ListenIP	Określ adres IP nasłuchiwania trappera.
ListenPort	Określ port nasłuchiwania trappera.
LoadModule	Określ moduł ładowany podczas uruchamiania proxy.
LoadModulePath	Określ pełną ścieżkę do lokalizacji modułów proxy.
LogFile	Określ plik dziennika proxy.
LogFileSize	Ustaw maksymalny rozmiar pliku dziennika.
LogRemoteCommands	Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
LogSlowQueries	Ustaw maksymalny czas trwania zapytania do bazy danych, po którego przekroczeniu zapytanie zostanie zarejestrowane (w milisekundach).
LogType	Określ typ wyjścia dziennika.
MaxConcurrentChecksPerPoller	Ustaw maksymalną liczbę asynchronicznych sprawdzeń, które mogą być jednocześnie wykonywane przez każdy poller HTTP agent, poller agent lub poller SNMP.
PidFile	Określ plik PID.
ProxyBufferMode	Określ mechanizm przechowywania danych historii, wykrywania i automatycznej rejestracji (dysk/pamięć/hybrydowy).
ProxyConfigFrequency	Ustaw częstotliwość pobierania danych konfiguracji z serwera Zabbix w sekundach.
ProxyLocalBuffer	Ustaw czas przechowywania danych lokalnie (w godzinach), nawet jeśli dane zostały już zsynchronizowane z serwerem.
ProxyMemoryBufferAge	Ustaw maksymalny wiek danych w buforze pamięci proxy w sekundach.
ProxyMemoryBufferSize	Ustaw rozmiar współdzielonej pamięci podręcznej dla zebranych danych historii, wykrywania i automatycznej rejestracji.
ProxyMode	Ustaw tryb działania proxy (aktywny/pasywny).
ProxyOfflineBuffer	Ustaw czas przechowywania danych (w godzinach) w przypadku braku połączenia z serwerem Zabbix.
Server	Jeśli ProxyMode jest ustawiony na `active`: określ adres IP lub nazwę DNS serwera Zabbix (address:port) albo klastra (address:port;address2:port), z którego mają być pobierane dane konfiguracji i do którego mają być wysyłane dane. Jeśli ProxyMode jest ustawiony na `passive`: określ adresy IP lub nazwy DNS serwera Zabbix.
SNMPTrapperFile	Określ plik tymczasowy używany do przekazywania danych z demona pułapek SNMP do proxy.
SocketDir	Określ katalog przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.
SourceIP	Określ źródłowy adres IP.
SSHKeyLocation	Określ lokalizację kluczy publicznych i prywatnych dla sprawdzeń i działań SSH.
SSLCertLocation	Określ lokalizację plików certyfikatów klienta SSL używanych do uwierzytelniania klienta.
SSLKeyLocation	Określ lokalizację plików kluczy prywatnych SSL używanych do uwierzytelniania klienta.
SSLCALocation	Określ lokalizację plików urzędów certyfikacji (CA) używanych do weryfikacji certyfikatu serwera SSL.
StartAgentPollers	Ustaw liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów agent.
StartAPMCollector	Ustaw wartość 1, aby uruchomić kolektor APM.
StartBrowserPollers	Ustaw liczbę wstępnie rozdzielonych instancji pollerów pozycji przeglądarki.
StartDBSyncers	Ustaw liczbę wstępnie rozdzielonych instancji synchronizatorów historii.
StartDiscoverers	Ustaw liczbę wstępnie rozdzielonych instancji procesów roboczych wykrywania.
StartHTTPAgentPollers	Ustaw liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów HTTP agent.
StartHTTPPollers	Ustaw liczbę wstępnie rozdzielonych instancji pollerów HTTP.
StartIPMIPollers	Ustaw liczbę wstępnie rozdzielonych instancji pollerów IPMI.
StartJavaPollers	Ustaw liczbę wstępnie rozdzielonych instancji pollerów Java.
StartODBCPollers	Ustaw liczbę wstępnie rozdzielonych instancji pollerów ODBC.
StartPingers	Ustaw liczbę wstępnie rozdzielonych instancji procesów pingujących ICMP.
StartPollersUnreachable	Ustaw liczbę wstępnie rozdzielonych instancji pollerów dla nieosiągalnych hostów (w tym IPMI i Java).
StartPollers	Ustaw liczbę wstępnie rozdzielonych instancji pollerów.
StartPreprocessors	Ustaw liczbę wstępnie uruchomionych instancji procesów roboczych przetwarzania wstępnego.
StartSNMPPollers	Ustaw liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów SNMP.
StartSNMPTrapper	Ustaw wartość ”1”, aby uruchomić proces trappera SNMP.
StartTelemetryQueryPollers	Ustaw liczbę wstępnie rozdzielonych instancji asynchronicznych pollerów zapytań telemetrycznych.
StartTrappers	Ustaw liczbę wstępnie rozdzielonych instancji trapperów.
StartVMwareCollectors	Ustaw liczbę wstępnie rozdzielonych instancji kolektorów VMware.
StatsAllowedIP	Określ adresy IP lub nazwy DNS zewnętrznych instancji Zabbix, które mogą żądać statystyk.
TelemetryProvider	Ustaw źródło danych telemetrycznych.

Parametr	Opis
Timeout	Określ czas oczekiwania (w sekundach) na ustanowienie połączenia i wymianę danych z proxy Zabbix, agent, usługą internetową oraz na sprawdzenia SNMP (z wyjątkiem pozycji SNMP `walk[OID]` i `get[OID]`).
TLSAccept	Określ wymagany poziom szyfrowania dla połączeń przychodzących z serwera Zabbix.
TLSCAFile	Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) służące do weryfikacji certyfikatu peera, używanej w szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Określ pełną ścieżkę do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów, używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Określ wymagany poziom szyfrowania dla połączeń z serwerem Zabbix.
TLSCRLFile	Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Określ pełną ścieżkę do pliku zawierającego klucz prywatny proxy, używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSListen	Wymagaj wyłącznie połączeń TLS na gnieździe trappera.
TLSPSKFile	Określ pełną ścieżkę do pliku zawierającego klucz wstępnie współdzielony proxy, używany w szyfrowanej komunikacji z serwerem Zabbix.
TLSPSKIdentity	Określ ciąg identyfikatora klucza wstępnie współdzielonego, używany w szyfrowanej komunikacji z serwerem Zabbix.
TLSServerCertIssuer	Określ dozwolonego wystawcę certyfikatu serwera.
TLSServerCertSubject	Określ dozwolony podmiot certyfikatu serwera.
TmpDir	Określ katalog tymczasowy.
TrapperTimeout	Określ limit czasu w sekundach dla: - pobierania danych konfiguracji z serwera Zabbix; - wykonywania skryptu globalnego lub zdalnego polecenia.
UnavailableDelay	Ustaw częstotliwość sprawdzania dostępności hosta w okresie niedostępności.
UnreachableDelay	Ustaw częstotliwość sprawdzania dostępności hosta w okresie nieosiągalności.
UnreachablePeriod	Ustaw, po ilu sekundach nieosiągalności host ma być traktowany jako niedostępny.
User	Zrezygnuj z uprawnień i używaj określonego, istniejącego użytkownika w systemie.
Vault	Określ dostawcę vault.
VaultAppRoleID	Określ identyfikator AppRole vault HashiCorp.
VaultAppSecretID	Określ identyfikator tajny AppRole vault HashiCorp.
VaultDBPath	Określ lokalizację pobierania poświadczeń bazy danych według kluczy.
VaultPrefix	Określ niestandardowy prefiks ścieżki lub zapytania vault.
VaultTLSCertFile	Określ nazwę pliku certyfikatu SSL używanego do uwierzytelniania klienta.
VaultTLSKeyFile	Określ nazwę pliku klucza prywatnego SSL używanego do uwierzytelniania klienta.
VaultToken	Określ token uwierzytelniania vault HashiCorp.
VaultURL	Określ adres URL serwera vault.
VMwareCacheSize	Ustaw rozmiar współdzielonej pamięci używanej do przechowywania danych VMware.
VMwareFrequency	Ustaw częstotliwość zbierania danych z pojedynczej usługi VMware w sekundach.
VMwarePerfFrequency	Ustaw częstotliwość pobierania statystyk liczników wydajności z pojedynczej usługi VMware w sekundach.
VMwareTimeout	Ustaw maksymalną liczbę sekund, przez które `vmware collector` będzie oczekiwać na odpowiedź usługi VMware.
WebDriverURL	Określ adres URL interfejsu WebDriver.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne demonów, a nie wartości w dostarczonych plikach konfiguracji.

- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracji wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametru

AllowRoot

Zezwalaj proxy na uruchamianie jako `root`. Jeśli ta opcja jest wyłączona, a proxy zostanie uruchomiony przez użytkownika `root`, proxy spróbuje zamiast tego przełączyć się na użytkownika `zabbix`. Nie ma to wpływu, jeśli zostanie uruchomiony przez zwykłego użytkownika.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

AllowUnsupportedDBVersions

Zezwalaj proxy na działanie z niewspieranymi wersjami bazy danych.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

APMListenIP

Określ adres IP, na którym kolektor APM powinien nasłuchiwać przychodzących danych telemetrycznych.
 Jeśli nie określono, kolektor APM będzie nasłuchiwać na wszystkich interfejsach sieciowych.

Wartość domyślna: `0.0.0.0`

APMListenPort

Określ port, na którym kolektor APM powinien nasłuchiwać przychodzących danych telemetrycznych.
 Domyślny port to standardowy port OpenTelemetry (OTLP/gRPC), używany domyślnie przez zestawy SDK i kolektory OpenTelemetry.

Wartość domyślna: 4317
 Zakres: 1024-32767

APMTLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) służące do weryfikowania certyfikatów aplikacji lub kolektorów OpenTelemetry wysyłających dane do kolektora APM.
 Jeśli ten parametr jest ustawiony, kolektor APM akceptuje wyłącznie szyfrowane połączenia; w przeciwnym razie akceptuje wyłącznie nieszyfrowane połączenia.

APMTLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat lub łańcuch certyfikatów kolektora APM.
 Ten parametr musi być ustawiony razem z `APMTLSKeyFile` i tylko wtedy, gdy ustawiono `APMTLSCAFile`.

APMTLSKeyFile

Określ pełną ścieżkę do pliku zawierającego klucz prywatny kolektora APM.
 Ten parametr musi być ustawiony razem z `APMTLSCertFile` i tylko wtedy, gdy ustawiono `APMTLSCAFile`.

CacheSize

Ustaw rozmiar pamięci podręcznej konfiguracji w bajtach. Rozmiar pamięci współdzielonej do przechowywania danych host i pozycji.

Domyślnie: 32M
 Zakres: 128K-64G

ConfigFrequency

Ten parametr jest **przestarzały** (zamiast niego użyj `ProxyConfigFrequency`).
Ustawia częstotliwość pobierania danych konfiguracyjnych z serwera Zabbix (w sekundach).
Parametr aktywnego proxy. Ignorowany dla pasywnych proxy (zobacz parametr `ProxyMode`).

Domyślnie: 3600
 Zakres: 1-604800

DataSenderFrequency

Ustaw częstotliwość wysyłania zebranych danych z proxy do serwera Zabbix. Należy pamiętać, że aktywny proxy nadal będzie odpytywać serwer Zabbix co sekundę w poszukiwaniu zadań zdalnych poleceń.
Parametr aktywnego proxy. Ignorowane dla pasywnych proxy (zobacz parametr `ProxyMode`).

Domyślnie: 1
 Zakres: 1-3600

DBHost

Określ host bazy danych (lub katalog gniazda).

Dla MySQL:

- `localhost` lub pusty ciąg używa domyślnego gniazda UNIX-domain.

Dla PostgreSQL:

- `localhost` jest rozwiązywany przez DNS (zwykle do `127.0.0.1`).
- Pusty ciąg używa domyślnego gniazda UNIX-domain.
- Ścieżka (na przykład `/var/run/pgbouncer`) powoduje użycie gniazda UNIX w tej ścieżce.
- Lista rozdzielana przecinkami może zawierać wiele wartości `host:port`. Każdy host jest sprawdzany po kolei, aż zostanie nawiązane połączenie do odczytu i zapisu. Przykład: `DBHost=localhost:5431,127.0.0.1:20051,zabbix.domain,[::1]:3005`

Domyślnie: `localhost`

DBName

Określ nazwę bazy danych lub ścieżkę do pliku bazy danych SQLite3 (wieloprocesowa architektura Zabbix nie pozwala na używanie bazy danych w pamięci, np. `:memory:`, `file::memory:?cache=shared` lub `file:memdb1?mode=memory&cache=shared`).
Ostrzeż
Nie próbuj używać tej samej bazy danych, z której korzysta serwer Zabbix.

Wymagane: Tak

DBPassword

Określ hasło do bazy danych. Zakomentuj tę linię, jeśli nie jest używane hasło. Ignorowane dla SQLite.

DBPort

Określ port hosta bazy danych Zabbix (zobacz DBHost). Ignorowane dla SQLite.
 DBSocket i DBPort wzajemnie się wykluczają w konfiguracji proxy. Określ tylko jeden z nich albo pozostaw oba niezdefiniowane.

Domyślnie dla MySQL: 3306

Domyślnie dla PostgreSQL: 5432

Zakres: 1024-65535

DBSchema

Określ nazwę schematu bazy danych. Używane dla PostgreSQL.

DBSocket

Określ ścieżkę do pliku gniazda MySQL.
DBSocket i DBPort wzajemnie się wykluczają w konfiguracji proxy. Określ tylko jeden z nich albo pozostaw oba niezdefiniowane.

DBUser

Określ użytkownika bazy danych. Ignorowane dla SQLite.

DBTLSConnect

Określ wartość, aby wymusić połączenie TLS z bazą danych: - *required* - połącz przy użyciu TLS; - *verify_ca* - połącz przy użyciu TLS i zweryfikuj certyfikat; - *verify_full* - połącz przy użyciu TLS, zweryfikuj certyfikat i sprawdź, czy tożsamość bazy danych określona przez DBHost jest zgodna z jej certyfikatem
W MySQL od wersji 5.7.11 oraz w PostgreSQL obsługiwane są następujące wartości: ”required”, ”verify”, ”verify_full”.
W MariaDB od wersji 10.2.6 obsługiwane są wartości ”required” i ”verify_full”.
Domyślnie nie jest ustawione żadna opcja, a zachowanie zależy od konfiguracji bazy danych.

DBTLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu bazy danych.

Obowiązkowe: nie (tak, jeśli DBTLSConnect jest ustawione na *verify_ca* lub *verify_full*)

DBTLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat proxy Zabbix do uwierzytelniania w bazie danych.

DBTLSKeyFile

Określ pełną ścieżkę do pliku zawierającego klucz prywatny używany do uwierzytelniania w bazie danych.

DBTLSCipher

Określ listę szyfrów szyfrowania, które proxy Zabbix dopuszcza dla protokołów TLS do wersji TLS v1.2 włącznie. Obsługiwane tylko dla MySQL.

DBTLSCipher13

Określ listę zestawów szyfrów szyfrowania, które proxy Zabbix dopuszcza dla protokołu TLS v1.3. Obsługiwane tylko dla MySQL, począwszy od wersji 8.0.16.

DebugLevel

Ustaw poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

EnableRemoteCommands

Zezwalaj na zdalne polecenia z serwera Zabbix.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

ExternalScripts

Określ lokalizację zewnętrznych skryptów (zależy od zmiennej instalacyjnej `datadir` ustawianej w czasie kompilacji).

Domyślnie: `/usr/local/share/zabbix/externalscripts`

Fping6Location

Określ lokalizację fping6. Upewnij się, że plik binarny fping6 ma właściciela root oraz ustawiony bit SUID. Pozostaw puste (`Fping6Location=`), jeśli narzędzie `fping` obsługuje adresy IPv6.

Domyślnie: `/usr/sbin/fping6`

FpingLocation

Określ lokalizację fping. Upewnij się, że plik binarny fping ma właściciela root oraz ustawiony bit SUID.

Domyślnie: `/usr/sbin/fping`

HistoryCacheSize

Ustaw rozmiar pamięci podręcznej historii w bajtach. Rozmiar pamięci współdzielonej do przechowywania danych historii.

Domyślnie: `16M`
 Zakres: 128K-16G

HistoryIndexCacheSize

Ustaw rozmiar pamięci podręcznej indeksu historii w bajtach. Rozmiar pamięci współdzielonej do indeksowania danych historii przechowywanych w pamięci podręcznej historii. Rozmiar pamięci podręcznej indeksu wymaga w przybliżeniu 100 bajtów na zbuforowanie jednej pozycji.

Domyślnie: `4M`
 Zakres: 128K-16G

Nazwa hosta

Określ unikalną nazwę proxy z uwzględnieniem wielkości liter. Upewnij się, że nazwa proxy jest znana serwerowi.
Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Określ pozycję do ustawienia `Hostname` (jeśli nie jest zdefiniowany).
 Ta pozycja będzie uruchamiana na proxy podobnie jak na agent.
 Ignorowane, jeśli `Hostname` jest ustawiony.
Nie obsługuje parametrów użytkownika, liczników wydajności ani aliasów; obsługuje `system.run[]`.

Domyślnie: `system.hostname`

HousekeepingFrequency

Ustaw częstotliwość procedury housekeeping (w godzinach). Housekeeping polega na usuwaniu nieaktualnych informacji z bazy danych.
Uwaga: Aby zmniejszyć obciążenie podczas uruchamiania proxy, housekeeping jest odraczany o 30 minut po starcie proxy. W związku z tym, jeśli `HousekeepingFrequency` ma wartość 1, pierwsza procedura housekeeping po uruchomieniu proxy zostanie wykonana po 30 minutach, a następnie będzie uruchamiana co godzinę.
Można wyłączyć automatyczny housekeeping, ustawiając HousekeepingFrequency na 0. W takim przypadku procedurę housekeeping można uruchomić wyłącznie za pomocą opcji sterowania w czasie działania `housekeeper_execute`.
Zobacz także opcje runtime control oraz szczegóły dotyczące housekeeping procedure.

Domyślnie: 1
 Zakres: 0-24

Include

Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracyjnego.
 Podczas instalacji Zabbixa ten parametr jest ustawiany na `/usr/local/etc`, chyba że zmienisz to podczas kompilacji.

Ogólne zasady dotyczące parametru Include:

- Ścieżka może być względna względem lokalizacji pliku zabbix_proxy.conf.
- W jednym pliku konfiguracyjnym można określić wiele parametrów Include.
- Zabbix przetwarza parametr Include natychmiast po jego odczytaniu, więc warto umieścić go na końcu pliku konfiguracyjnego.

Zasady dotyczące dołączanych plików:

- Każdy plik musi być możliwy do odczytu.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie uruchomi się.

Zasady dotyczące dołączania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności dołączania (na przykład pliki nie są dołączane alfabetycznie).
- Nie definiuj tego samego parametru w więcej niż jednym dołączonym pliku (na przykład, aby zastąpić ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Uważaj na kopie zapasowe tworzone automatycznie przez niektóre edytory tekstu (na przykład edycja include/my_specific.conf może utworzyć include/my_specific.conf.BAK), ponieważ oba pliki zostaną dołączone.

Aby dołączyć tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (*) do dopasowywania wzorca.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

JavaGateway

Określ adres IP lub nazwę hosta Zabbix Java gateway. Wymagane tylko wtedy, gdy uruchomione są pollery Java.

JavaGatewayPort

Określa port Zabbix Java gateway.

Domyślnie: 10052
 Zakres: 1024-32767

ListenBacklog

Ustawia maksymalną liczbę oczekujących połączeń w kolejce TCP.
Wartość domyślna jest stałą zakodowaną na stałe i zależy od systemu.
Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać bez ostrzeżenia obcięte do „maksimum określonego przez implementację”.

Domyślnie: SOMAXCONN
 Zakres: 0 - INT_MAX

ListenIP

Określ adres IP nasłuchiwania trappera. Lista rozdzielona przecinkami.
Trapper będzie nasłuchiwał na wszystkich interfejsach sieciowych, jeśli ten parametr nie zostanie określony.

Domyślnie: 0.0.0.0

ListenPort

Określ port nasłuchiwania trappera.

Domyślnie: 10051
 Zakres: 1024-32767

LoadModule

Określ moduł, który ma zostać załadowany podczas uruchamiania proxy. Moduły służą do rozszerzania funkcjonalności proxy. Moduł musi znajdować się w katalogu określonym przez LoadModulePath lub przed nazwą modułu należy podać ścieżkę. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od /), parametr LoadModulePath jest ignorowany.
 Formaty: - LoadModule=<module.so> - LoadModule=<path/module.so> - LoadModule=</abs_path/module.so>

Można określić wiele parametrów LoadModule.

LoadModulePath

Pełna ścieżka do lokalizacji modułów proxy. Domyślna wartość zależy od opcji kompilacji.

LogFile

Określ plik dziennika proxy.

Obowiązkowe: Tak, jeśli LogType jest ustawione na file; w przeciwnym razie nie

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
0 - wyłącza automatyczną rotację dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku z jakiegokolwiek powodu się nie powiedzie, istniejący plik dziennika zostanie skrócony i utworzony od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

LogSlowQueries

Ustaw maksymalny czas trwania zapytania do bazy danych, po którym zostanie ono zapisane w logu (w milisekundach).
0 - nie rejestruj wolnych zapytań.
Ta opcja staje się dostępna od `DebugLevel=3`.

Domyślnie: 0
 Zakres: 0-3600000

LogType

Określ typ wyjścia logu: - *file* - zapisuj log do pliku określonego parametrem `LogFile`; - *system* - zapisuj log do syslog; - *console* - zapisuj log do standardowego wyjścia.

Domyślnie: `file`

MaxConcurrentChecksPerPoller

Ustaw maksymalną liczbę asynchronicznych kontroli, które mogą być wykonywane jednocześnie przez każdy poller HTTP agent, poller agent lub poller SNMP. Zobacz `StartHTTPAgentPollers`, `StartAgentPollers` i `StartSNMPPollers`.

Domyślnie: 1000
 Zakres: 1-1000

PidFile

Określa plik PID.

Domyślnie: `/tmp/zabbix_proxy.pid`

ProxyBufferMode

Określa mechanizm przechowywania danych historycznych, wykrywania i automatycznej rejestracji: - *disk* - dane są przechowywane w bazie danych i wysyłane z bazy danych; - *memory* - dane są przechowywane w pamięci i wysyłane z pamięci. Jeśli bufor wyczerpie pamięć, stare dane zostaną odrzucone. Podczas zamykania bufor jest usuwany. - *hybrid* - bufor proxy zwykle działa jak w trybie pamięci, dopóki nie wyczerpie pamięci lub najstarszy rekord nie przekroczy skonfigurowanego wieku. Jeśli tak się stanie, bufor jest opróżniany do bazy danych i działa jak w trybie dyskowym, dopóki wszystkie dane nie zostaną wysłane, po czym ponownie zaczyna działać z pamięcią. Podczas zamykania bufor pamięci jest opróżniany do bazy danych.

Zobacz także: Proxy memory buffer.

Domyślnie: `disk`
 Wartości: disk; memory; hybrid

ProxyConfigFrequency

Ustawia częstotliwość pobierania danych konfiguracyjnych z serwera Zabbix w sekundach.
Parametr aktywnego proxy. Ignorowany dla pasywnych proxy (zobacz parametr `ProxyMode`).

Domyślnie: 10
 Zakres: 1-604800

ProxyLocalBuffer

Ustaw czas przechowywania danych lokalnie (w godzinach), nawet jeśli dane zostały już zsynchronizowane z serwerem..
Ten parametr może być używany, jeśli dane lokalne będą wykorzystywane przez aplikacje firm trzecich.

Domyślnie: 0
 Zakres: 0-720

ProxyMemoryBufferAge

Ustaw maksymalny wiek danych w buforze pamięci proxy w sekundach.
 Gdy opcja jest włączona (wartość różna od zera), a rekordy w buforze pamięci proxy są starsze, wymusza to przełączenie bufora proxy w tryb bazy danych, aż wszystkie rekordy zostaną wysłane do serwera. Ten parametr musi być mniejszy lub równy parametrowi `ProxyOfflineBuffer`.

Domyślnie: 0
 Zakres: 0;600-864000

ProxyMemoryBufferSize

Ustaw rozmiar współdzielonej pamięci podręcznej dla zbieranych danych historycznych, wykrywania i automatycznej rejestracji (w bajtach). Jeśli włączone (wartość różna od zera), proxy będzie przechowywać w pamięci dane historyczne, wykrywania i automatycznej rejestracji, dopóki pamięć podręczna nie zostanie zapełniona lub zapisane rekordy nie będą starsze niż zdefiniowane `ProxyMemoryBufferAge`. Tego parametru nie można używać razem z parametrem `ProxyLocalBuffer`.

Domyślnie: 0
 Zakres: 0;128K-2G

ProxyMode

Ustaw tryb pracy proxy.
0 - proxy w trybie aktywnym
1 - proxy w trybie pasywnym
Uwaga konfiguracja proxy (wrażliwe dane) może stać się dostępna dla stron mających dostęp do portu trapper serwera Zabbix podczas korzystania z aktywnego proxy. Jest to możliwe, ponieważ każdy może podszyć się pod aktywne proxy i zażądać danych konfiguracyjnych; uwierzytelnianie nie jest przeprowadzane.

Domyślnie: 0
 Zakres: 0-1

ProxyOfflineBuffer

Ustaw czas przechowywania danych (w godzinach) w przypadku braku łączności z serwerem Zabbix.
Starsze dane zostaną utracone.

Domyślnie: 1
 Zakres: 1-720

Serwer

Jeśli ProxyMode jest ustawiony na `active`: określ adres IP serwera Zabbix lub nazwę DNS (address:port) albo klaster (address:port;address2:port), z którego mają być pobierane dane konfiguracyjne i do którego mają być wysyłane dane.
Jeśli port nie zostanie określony, zostanie użyty port domyślny.
Węzły klastra muszą być oddzielone średnikiem.

Jeśli ProxyMode jest ustawiony na `passive`: określ adresy IP, opcjonalnie w notacji CIDR, lub nazwy DNS serwera Zabbix. Lista rozdzielana przecinkami.
 Połączenia przychodzące będą akceptowane tylko z adresów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo.
`::/0` umożliwi dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Obowiązkowe: tak

SNMPTrapperFile

Określ plik tymczasowy używany do przekazywania danych z demona pułapek SNMP do proxy.
Musi być taki sam jak w pliku `zabbix_trap_receiver.pl` lub w pliku konfiguracji SNMPTT.

Domyślnie: `/tmp/zabbix_traps.tmp`

SocketDir

Określa katalog do przechowywania gniazd IPC używanych przez wewnętrzne usługi Zabbix.

Domyślnie: `/tmp`

SourceIP

Określ adres IP źródłowy dla:

- połączeń wychodzących do serwera Zabbix
- połączeń bezagentowych (VMware, SSH, JMX, SNMP, Telnet i proste sprawdzenia)
- połączeń HTTP agent
- żądań HTTP JavaScript w pozycji skryptowej
- żądań HTTP JavaScript w przetwarzaniu wstępnym
- połączeń do Vault

SSHKeyLocation

Określa lokalizację kluczy publicznych i prywatnych dla sprawdzeń SSH oraz działań.

SSLCertLocation

Określ lokalizację plików certyfikatu klienta SSL używanych do uwierzytelniania klienta.
Ten parametr jest używany w monitorowaniu sieci web, do komunikacji z Vault oraz dla źródła danych telemetrycznych ClickHouse.

SSLKeyLocation

Określ lokalizację plików kluczy prywatnych SSL używanych do uwierzytelniania klienta.
Ten parametr jest używany w monitorowaniu sieci web, do komunikacji z Vault oraz dla źródła danych telemetrycznych ClickHouse.

SSLCALocation

Określ lokalizację plików urzędu certyfikacji (CA) używanych do weryfikacji certyfikatu serwera SSL.
Pamiętaj, że wartość tego parametru zostanie ustawiona jako opcja CURLOPT_CAPATH biblioteki libcurl. W przypadku wersji libcurl wcześniejszych niż 7.42.0 działa to tylko wtedy, gdy libcurl została skompilowana z użyciem OpenSSL. Więcej informacji znajdziesz na stronie internetowej cURL.
Ten parametr jest używany w monitoringu sieci WWW, uwierzytelnianiu SMTP, pozycjach HTTP agent, do komunikacji z Vault oraz dla źródła danych telemetrycznych ClickHouse.

StartAgentPollers

Ustawia liczbę wstępnie rozwidlonych instancji Zabbix agent `pollers`. Zobacz `MaxConcurrentChecksPerPoller`.

Domyślnie: 1
 Zakres: 0-1000

StartAPMCollector

Ustaw wartość 1, aby uruchomić wstępnie rozwidlone wystąpienie kolektora APM (procesy `apm manager` i `apm worker`).

Wartość domyślna: 0
 Zakres: 0-1

StartBrowserPollers

Ustawia liczbę wcześniej rozwidlonych instancji `pollers` dla pozycji przeglądarki `pollers`.

Domyślnie: 1
 Zakres: 0-1000

StartDBSyncers

Ustaw liczbę wstępnie rozwidlonych instancji `history syncers`.
Uwaga: Zachowaj ostrożność podczas zmiany tej wartości, ponieważ jej zwiększenie może przynieść więcej szkody niż pożytku.

Domyślnie: 4
 Zakres: 1-100

StartDiscoverers

Ustaw liczbę wstępnie rozwidlonych instancji `discovery workers`.

Domyślnie: 5
 Zakres: 0-1000

StartHTTPAgentPollers

Ustaw liczbę wstępnie rozwidlonych instancji `HTTP agent pollers`. Zobacz `MaxConcurrentChecksPerPoller`.

Domyślnie: 1
 Zakres: 0-1000

StartHTTPPollers

Ustaw liczbę wstępnie utworzonych instancji `HTTP pollers`.

Domyślnie: 1
 Zakres: 0-1000

StartIPMIPollers

Ustaw liczbę wstępnie utworzonych instancji `IPMI pollers`.

Domyślnie: 0
 Zakres: 0-1000

StartJavaPollers

Ustaw liczbę wstępnie rozwidlonych instancji `Java pollers`.

Domyślnie: 0
 Zakres: 0-1000

StartODBCPollers

Ustaw liczbę wstępnie rozwidlonych instancji `ODBC pollers`.

Domyślnie: 1
 Zakres: 0-1000

StartPingers

Ustaw liczbę wstępnie rozwidlonych instancji `ICMP pingers`.

Domyślnie: 1
 Zakres: 0-1000

StartPollersUnreachable

Ustawia liczbę wcześniej utworzonych instancji pollerów dla niedostępnych hostów (w tym IPMI i Java). Co najmniej jeden poller dla niedostępnych hostów musi być uruchomiony, jeśli uruchomione są zwykłe pollery, pollery IPMI lub Java.

Domyślnie: 1
 Zakres: 0-1000

StartPollers

Ustaw liczbę wstępnie utworzonych instancji pollers.

Domyślnie: 5
 Zakres: 0-1000

StartPreprocessors

Ustaw liczbę wcześniej uruchomionych instancji wątków przetwarzania wstępnego worker na wartość nie mniejszą niż liczba dostępnych rdzeni CPU. Należy ustawić więcej workerów, jeśli przetwarzanie wstępne nie jest ograniczone przez CPU i generuje wiele żądań sieciowych.

Domyślnie: 16
 Zakres: 1-1000

StartSNMPPollers

Ustaw liczbę wcześniej utworzonych instancji SNMP pollers. Zobacz MaxConcurrentChecksPerPoller.

Domyślnie: 1
 Zakres: 0-1000

StartTelemetryQueryPollers

Ustaw liczbę wstępnie utworzonych instancji asynchronicznych pollerów zapytań Telemetry. Zobacz MaxConcurrentChecksPerPoller.

Wartość domyślna: 1
 Zakres: 0-1000

StartSNMPTrapper

Ustaw na "1", aby uruchomić proces SNMP trapper.

Domyślnie: 0
 Zakres: 0-1

StartTrappers

Ustaw liczbę wcześniej utworzonych instancji trappers.
Trappers akceptują przychodzące połączenia od Zabbix sender i aktywnych agentów.

Domyślnie: 5
 Zakres: 0-1000

StartVMwareCollectors

Ustaw liczbę wstępnie rozwidlonych instancji vmware collector.

Domyślnie: 0
 Zakres: 0-250

StatsAllowedIP

Określ adresy IP (opcjonalnie w notacji CIDR) lub nazwy DNS zewnętrznych instancji Zabbix, które mają prawo żądać statystyk. Lista rozdzielana przecinkami.
 Jeśli ten parametr nie jest ustawiony, żadne żądania statystyk nie będą akceptowane.
Jeśli obsługa IPv6 jest włączona, to `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4.

Przykład:

```
StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

TelemetryProvider

Ustaw źródło danych telemetrycznych w formacie `<provider>[;<options>]`, gdzie:

- `<provider>` - jedno z obsługiwanych źródeł danych telemetrycznych: `clickhouse`
- `<options>` - rozdzielane przecinkami pary `key=value`. Wartości można ujmować w cudzysłowy.

Opcje dla ClickHouse:

- `url` - ustaw adres URL HTTP lub HTTPS używany do łączenia ze źródłem danych telemetrycznych.
- `db` - ustaw nazwę bazy danych.
- `username` - ustaw nazwę użytkownika używaną do logowania do bazy danych.
- `password` - ustaw hasło używane do logowania do bazy danych.
- `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_cert_location`, `ssl_key_location` - ustaw te opcje, aby skonfigurować bezpieczne połączenie (TLS/SSL) z bazą danych.
- `source_ip` - ustaw źródłowy adres IP używany dla połączeń wychodzących ze źródłem danych telemetrycznych.
- `vault_path` - ustaw ścieżkę do sekretu Vault zawierającego wartości `username` i `password`. Nie używaj razem z opcjami `username` i `password`.

Przykład dla ClickHouse:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,username=zabbix,password=zabbix
```

Przykład dla ClickHouse z Vault:

```
TelemetryProvider=clickhouse;url=http://localhost:8123,db=zabbix,vault_path=clickhouse
```

Timeout

Określ, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z serwerem Zabbix, agent, usługą webową oraz starszymi sprawdzeniami SNMP (pojedynczy numer OID lub ciąg znaków).

Ten parametr określa czas trwania różnych operacji komunikacyjnych:

- Zdalne wykonywanie poleceń na agent Zabbix
- Wykonywanie poleceń SSH / Telnet
- Ponowne planowanie pozycji, gdy interfejs IPMI staje się niedostępny
- Wysyłanie odpowiedzi do serwera Zabbix w przypadku niepowodzenia wymiany danych z powodu problemów z uprawnieniami lub szyfrowaniem
- Termin dla asynchronicznych gniazd IPC i opcji sterowania w czasie wykonywania
- Asynchroniczne zapytania DNS pollera
- Odpowiedź dla heartbeat aktywnego sprawdzenia
- Pobieranie danych (wartości) agenta Zabbix z aktywnych agent
- Pobieranie danych z Zabbix sender
- Wysyłanie listy aktywnych sprawdzeń do agent Zabbix

Ten timeout **nie** będzie używany dla tych sprawdzeń, dla których skonfigurowano ustawienia flexible timeout w frontend (na poziomie globalnym, proxy lub dla pojedynczej pozycji). Na przykład pozycje SNMP `walk[OID]` i `get[OID]` używają skonfigurowanego timeout z frontend; starsze sprawdzenia SNMP nadal używają wartości timeout serwera.

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Określ poziom szyfrowania wymagany dla połączeń przychodzących z serwera Zabbix. Dotyczy proxy pasywnego, jest ignorowane w przypadku proxy aktywnego. Można podać wiele wartości, oddzielając je przecinkami: - *unencrypted* - akceptuj połączenia bez szyfrowania (domyślnie); - *psk* - akceptuj połączenia z TLS i wstępnie współdzielonym kluczem (PSK); - *cert* - akceptuj połączenia z TLS i certyfikatem.

Obowiązkowe: tak dla proxy pasywnego, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych CA do weryfikacji certyfikatu peer, używanego do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat proxy lub łańcuch certyfikatów, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSCipherAll

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

TLSCipherPSK

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Określa poziom szyfrowania wymagany dla połączeń proxy z serwerem Zabbix. Używane dla aktywnego proxy, ignorowane dla pasywnego proxy. Można określić tylko jedną wartość: - *unencrypted* - połączenie bez szyfrowania (domyślnie); - *psk* - połączenie z użyciem TLS i współdzielonego klucza wstępnego (PSK); - *cert* - połączenie z użyciem TLS i certyfikatu.

Wymagane: tak dla aktywnego proxy, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz proxy, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSListen

Wymagaj połączeń wyłącznie TLS na gnieździe trapper.

Obsługiwane wartości:

- required - akceptuj tylko połączenia TLS

TLSPSKFile

Podaj pełną ścieżkę do pliku zawierającego współdzielony klucz proxy, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSPSKIdentity

Określ ciąg tożsamości pre-shared key, używany do szyfrowanej komunikacji z serwerem Zabbix.

TLSServerCertIssuer

Określ dozwolonego wystawcę certyfikatu serwera.

TLSServerCertSubject

Określ dozwolony subject certyfikatu serwera.

TmpDir

Określ katalog tymczasowy.

Domyślnie: `/tmp`

TrapperTimeout

Określ limit czasu w sekundach dla:
 - pobierania danych konfiguracyjnych z serwera Zabbix;
 - wykonywania globalnego skryptu lub zdalnego polecenia.

Domyślnie: 300
 Zakres: 1-300

UnavailableDelay

Ustaw częstotliwość sprawdzania dostępności hosta w okresie niedostępności w sekundach.

Domyślnie: 60
 Zakres: 1-3600

UnreachableDelay

Ustaw częstotliwość sprawdzania dostępności hosta podczas okresu niedostępności w sekundach.

Domyślnie: 15
 Zakres: 1-3600

UnreachablePeriod

Ustawia, po ilu sekundach niedostępności uznać host za niedostępny.

Domyślnie: 45
 Zakres: 1-3600

Użytkownik

Przełącz uprawnienia na konkretnego, istniejącego użytkownika w systemie.
Ma to efekt tylko wtedy, gdy jest uruchamiane jako `root` i `AllowRoot` jest wyłączone.

Domyślnie: `zabbix`

Vault

Określ dostawcę vault: - *HashiCorp* - HashiCorp KV Secrets Engine w wersji 2; - *CyberArk* - CyberArk Central Credential Provider
Musi być zgodny z dostawcą vault ustawionym w frontend.

Domyślnie: `HashiCorp`

VaultAppRoleID

Określ identyfikator AppRole usługi HashiCorp Vault.
 Jeśli ustawione, włącza uwierzytelnianie AppRole. Nie można używać razem z VaultToken.
 Obsługiwane tylko dla HashiCorp Vault.

VaultAppSecretID

Określ identyfikator tajny AppRole w HashiCorp Vault.
 Obsługiwane tylko dla HashiCorp Vault.

VaultDBPath

Określa lokalizację pobierania poświadczeń bazy danych według kluczy. Jest to ścieżka lub zapytanie Vault, zależnie od używanego Vault.

Klucze używane przez HashiCorp to `password` i `username`.

Przykładowa ścieżka z `VaultPrefix=/v1/secret/data/zabbix/`:

```
database
```

Przykładowa ścieżka bez VaultPrefix:

```
secret/zabbix/database
```

Klucze używane przez CyberArk to `Content` i `UserName`.

Przykład:

```
AppID=zabbix_server&Query=Safe=passwordSafe;Object=zabbix_proxy_database
```

Tę opcję można użyć tylko wtedy, gdy `DBUser` i `DBPassword` nie są określone.

VaultPrefix

Określ niestandardowy prefiks dla ścieżki lub zapytania vault, w zależności od Vault. Jeśli nie zostanie określony, zostaną użyte najbardziej odpowiednie wartości domyślne.
Uwaga: dla HashiCorp po punkcie montowania automatycznie dodawany jest ciąg 'data', jeśli `VaultPrefix` nie jest określony.
Uwaga: dla HashiCorp po punkcie montowania automatycznie dodawany jest ciąg 'data', jeśli `VaultPrefix` nie jest określony.

Przykładowy prefiks dla HashiCorp:

```
/v1/secret/data/zabbix/
```

Przykładowy prefiks dla Cyberark:

```
/AIMWebService/api/Accounts?
```

VaultTLSCertFile

Określ nazwę pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM1.
Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole pliku klucza SSL puste.
Katalog zawierający ten plik jest określany przez parametr konfiguracyjny SSLCertLocation.
Ta opcja może zostać pominięta, ale jest zalecana dla vault CyberArkCCP.

VaultTLSKeyFile

Określ nazwę pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik prywatnego klucza musi być w formacie PEM1.
Katalog zawierający ten plik jest określany przez parametr konfiguracyjny SSLKeyLocation.
Opcję tę można pominąć, ale jest zalecana dla vault CyberArkCCP.

VaultToken

Określ token uwierzytelniania HashiCorp Vault, który powinien zostać wygenerowany wyłącznie dla proxy Zabbix z uprawnieniami tylko do odczytu do ścieżki określonej w opcjonalnym parametrze konfiguracyjnym VaultDBPath.
Wystąpi błąd, jeśli `VaultToken` i zmienna środowiskowa `VAULT_TOKEN` są zdefiniowane jednocześnie.
 Nie można używać z VaultAppRoleID.
 Obsługiwane tylko dla HashiCorp Vault.

VaultURL

Określ adres URL serwera vault. Katalog certyfikatów CA obowiązujący w całym systemie zostanie użyty, jeśli nie zostanie określony SSLCALocation.

Domyślnie: `https://127.0.0.1:8200`

VMwareCacheSize

Ustaw rozmiar pamięci współdzielonej do przechowywania danych VMware.
Do monitorowania użycia pamięci podręcznej VMware można użyć wewnętrznego sprawdzenia `zabbix[vmware,buffer,...]` (zobacz Wewnętrzne sprawdzenia).
Należy pamiętać, że pamięć współdzielona nie jest przydzielana, jeśli nie skonfigurowano żadnych instancji kolektora vmware do uruchomienia.

Domyślnie: 8M
 Zakres: 256K-2G

VMwareFrequency

Ustaw częstotliwość zbierania danych z pojedynczej usługi VMware (w sekundach).
Ta częstotliwość powinna być ustawiona na najmniejszy interwał aktualizacji dowolnej pozycji monitorującej VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwarePerfFrequency

Ustaw częstotliwość pobierania statystyk liczników wydajności z pojedynczej usługi VMware (w sekundach).
Ta częstotliwość powinna być ustawiona na najmniejszy interwał aktualizacji dowolnej monitorowanej pozycja VMware, która używa liczników wydajności VMware.

Domyślnie: 60
 Zakres: 10-86400

VMwareTimeout

Ustaw maksymalną liczbę sekund, przez którą `vmware collector` będzie czekać na odpowiedź od usługi VMware (vCenter lub hipernadzorcy ESX).

Domyślnie: 10
 Zakres: 1-300

WebDriverURL

Określ adres URL interfejsu WebDriver.

Przykład (używany z samodzielnym serwerem Selenium WebDriver):

```
WebDriverURL=http://localhost:4444
```

Agent

Przegląd agent Zabbix jest wdrażany na monitorowanym celu w celu aktywnego monitorowania lokalnych zasobów i aplikacji (dysków twardych, pamięci, statystyk procesora itp.).

agent zbiera lokalnie informacje operacyjne i przekazuje dane do serwera Zabbix w celu dalszego przetwarzania. W przypadku awarii (takiej jak całkowite zapełnienie dysku twardego lub zatrzymanie procesu usługi) serwer Zabbix może aktywnie powiadomić administratorów konkretnej maszyny, która zgłosiła awarię.

agenty Zabbix są bardzo wydajne dzięki wykorzystaniu natywnych wywołań systemowych do zbierania informacji statystycznych.

Na jednym hoście można uruchomić wiele instancji agenta.

Zobacz także:

- Protokoły sprawdzania pasywnego i aktywnego
- Uruchamianie agenta jako root
- Parametry konfiguracji: agent Zabbix
- Parametry konfiguracji: agent Zabbix (Windows)
- agent Zabbix w systemie Microsoft Windows
- agent Zabbix w systemie IBM AIX

Kontrole pasywne i aktywne

Agenty Zabbix mogą wykonywać kontrole pasywne i aktywne:

- Kontrole pasywne - agent Zabbix odpowiada na żądanie z serwera Zabbix (lub proxy). Na przykład serwer pobiera dane (np. obciążenie CPU), a agent zwraca wynik.
- Kontrole aktywne - agent Zabbix zbiera i wysyła dane bez czekania na żądanie z serwera Zabbix (lub proxy). Najpierw pobiera listę pozycji monitorowania z serwera (obciążenie CPU, dostępna pamięć itp.), a następnie zbiera wymagane dane i okresowo wysyła do niego nowe wartości.

Typ kontroli agenta jest konfigurowany przez wybranie odpowiedniego typu pozycji monitorowania. Agent Zabbix przetwarza pozycje typu *Zabbix agent* lub *Zabbix agent (active)*.

Obsługiwane platformy Informacje o obsługiwanych platformach znajdują się na stronie Wymagania.

Agent w systemach uniksopodobnych Agent Zabbixa w systemach uniksopodobnych jest uruchamiany na hostcie, który jest monitorowany.

Instalacja

agent Zabbix można zainstalować w systemach opartych na systemie Linux, korzystając z jednej z następujących metod:

- Pakiety Zabbix - wybierz komponent Agent (po wybraniu wersji Zabbix, dystrybucji systemu operacyjnego i wersji systemu operacyjnego), a następnie postępuj zgodnie z podanymi instrukcjami;
- Źródła Zabbix - pobierz pliki źródłowe i skompiluj agent Zabbix, konfigurując źródła z opcją `--enable-agent`.

Attention:
Ogólnie rzecz biorąc, 32-bitowe agenty Zabbix będą działać w systemach 64-bitowych, ale w niektórych przypadkach mogą wystąpić problemy.

Note:
Wstępnie skompilowane pliki binarne agent Zabbix są dostępne do pobrania dla systemów macOS, IBM AIX, FreeBSD, OpenBSD i Solaris. Starsze pliki binarne, zgodne z bieżącą wersją serwera/proxy Zabbix, są dostępne dla systemów NetBSD i HP-UX. Informacje dotyczące instalowania i uruchamiania agent w systemie IBM AIX można znaleźć na stronie agent Zabbix w systemie IBM AIX.

Jeśli zainstalowano jako pakiet

Agent Zabbix działa jako proces demona. Agent można uruchomić, wykonując:

```
systemctl start zabbix-agent
```

To zadziała w większości systemów GNU/Linux. W innych systemach może być konieczne uruchomienie:

```
/etc/init.d/zabbix-agent start
```

Aby zatrzymać, uruchomić ponownie lub sprawdzić status agenta Zabbix, użyj następujących poleceń:

```
systemctl stop zabbix-agent
systemctl restart zabbix-agent
systemctl status zabbix-agent
```

Jeśli zainstalowano z archiwum binarnego

Po rozpakowaniu archiwum znajdź plik binarny `zabbix_agentd` oraz plik konfiguracyjny `zabbix_agentd.conf`. Ich lokalizacja w archiwum zależy od platformy (na przykład `sbin/` i `conf/` albo `usr/sbin/` i `etc/zabbix/`). Uruchom plik binarny bezpośrednio z rozpakowanego katalogu i określ plik konfiguracyjny za pomocą opcji `-c`. Na przykład:

```
/opt/zabbix/sbin/zabbix_agentd -c /opt/zabbix/etc/zabbix/zabbix_agentd.conf
```

W systemach uniksopodobnych agent Zabbix jest przeznaczony do uruchamiania jako użytkownik inny niż root. Jeśli zostanie uruchomiony jako root, przełącza się na użytkownika `zabbix` (lub użytkownika określonego przez parametr `[User](<PARAMS>#user)`), chyba że w pliku konfiguracyjnym ustawiono `[AllowRoot=1](<PARAMS>#allowroot)`. Ten użytkownik musi istnieć w systemie; w przeciwnym razie agent nie zostanie uruchomiony.

W systemie IBM AIX agent może być również uruchamiany za pomocą System Resource Controller (SRC) oraz podczas uruchamiania systemu; zobacz Agent Zabbix w systemie IBM AIX.

Uruchamianie ręczne

Możesz uruchomić agent Zabbix, uruchamiając bezpośrednio plik binarny `zabbix_agentd`, na przykład:

```
zabbix_agentd
```

Działa to, gdy katalog zawierający `zabbix_agentd` znajduje się w zmiennej `PATH`.

Agent w systemach Windows Agent Zabbix w systemie Windows działa jako usługa systemu Windows.

Instalacja

agent Zabbix można zainstalować w systemie Windows jedną z następujących metod:

- Wstępnie skompilowane binaria agenta Zabbix - pobierz pakiet instalacyjny MSI agenta Zabbix i postępuj zgodnie z instrukcjami na stronie Instalacja agenta Windows z pliku MSI;
- Źródła Zabbix - pobierz pliki źródłowe i postępuj zgodnie z instrukcjami na stronie Budowanie agenta Zabbix w systemie Windows.

Dodatkowe informacje na temat instalacji agenta Zabbix (z archiwum ZIP) jako usługi systemu Windows znajdują się na stronie agent Zabbix w Microsoft Windows.

Opcje Z agent można używać następujących parametrów wiersza poleceń:

Parametr	Opis
agent UNIX i Windows	
`-c --config <config-file>`	Ścieżka do pliku konfiguracyjnego. Tej opcji można użyć, aby określić plik konfiguracyjny inny niż domyślny.
`-f --foreground`	Uruchom agenta Zabbix na pierwszym planie (domyślnie: `true`).
`-p --print`	Wyświetl znane pozycje i zakończ działanie. Aby zwrócić również wyniki parametru użytkownika, należy określić plik konfiguracyjny (jeśli nie znajduje się on w domyślnej lokalizacji).
`-t --test <item key>`	Przetestuj określoną pozycję i zakończ działanie. Aby zwrócić również wyniki parametru użytkownika, należy określić plik konfiguracyjny (jeśli nie znajduje się on w domyślnej lokalizacji).
`-T --test-config`	Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.
`-h --help`	Wyświetl informacje pomocy.
`-V --version`	Wyświetl numer wersji.
Tylko agent UNIX	

Parametr	Opis
`-R --runtime-control <option>`	Wykonaj funkcje administracyjne. Zobacz sterowanie w czasie działania.
Tylko agent Windows	
`-m --multiple-agents`	Używaj wielu instancji agenta (z opcjami `-i`, `-d`, `-s`, `-x`). Aby rozróżnić nazwy usług poszczególnych instancji, każda nazwa usługi będzie zawierać wartość Hostname z określonego pliku konfiguracyjnego.
`-S --startup-type <value>`	Ustaw typ uruchamiania usługi agenta Zabbix dla systemu Windows. Dozwolone wartości: `automatic` - *(domyślnie)* automatycznie uruchom usługę podczas uruchamiania systemu Windows; `delayed` - opóźnij uruchomienie usługi do czasu zakończenia uruchamiania usług uruchamianych automatycznie (dostępne w systemach Windows Server 2008/Vista i nowszych); `manual` - uruchamiaj usługę ręcznie (przez użytkownika lub aplikację); `disabled` - wyłącz usługę, aby nie mogła zostać uruchomiona przez użytkownika ani aplikację. Tej opcji można użyć razem z opcją `-i` lub osobno, aby zmodyfikować typ uruchamiania już zainstalowanej usługi.
`-i --install`	Zainstaluj agenta Zabbix dla systemu Windows jako usługę.
`-d --uninstall`	Odinstaluj usługę agenta Zabbix dla systemu Windows.
`-s --start`	Uruchom usługę agenta Zabbix dla systemu Windows.
`-x --stop`	Zatrzymaj usługę agenta Zabbix dla systemu Windows.

Przykłady użycia parametrów wiersza poleceń:

- Wyświetlanie wszystkich wbudowanych pozycji agenta wraz z wartościami.
- Testowanie parametru użytkownika z kluczem `mysql.ping` zdefiniowanym w określonym pliku konfiguracyjnym.
- Instalowanie usługi agenta Zabbix dla systemu Windows przy użyciu domyślnej ścieżki do pliku konfiguracyjnego.
- Instalowanie usługi agenta Zabbix [Hostname] dla systemu Windows przy użyciu pliku konfiguracyjnego znajdującego się w tym samym folderze co plik wykonywalny agenta oraz nadanie nazwie usługi unikatowego rozszerzenia na podstawie wartości `Hostname` z pliku konfiguracyjnego.
- Modyfikowanie typu uruchamiania zainstalowanej usługi agenta Zabbix dla systemu Windows przy użyciu pliku konfiguracyjnego znajdującego się w tym samym folderze co plik wykonywalny agenta.

```
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
```

Sterowanie w czasie działania

Za pomocą opcji sterowania w czasie działania można zmieniać poziom rejestrowania procesów agent.

Option	Description	Target
`log_level_increase[<target>]`	Zwiększ poziom rejestrowania. Jeśli element docelowy nie zostanie określony, dotyczy to wszystkich procesów.	Element docelowy można określić jako: **typ procesu** - wszystkie procesy określonego typu (np. `listener`) Zobacz wszystkie typy procesów agent. **typ procesu,N** - typ i numer procesu (np. `listener,3`) **pid** - identyfikator procesu (1 do 65535). W przypadku większych wartości należy określić element docelowy jako „typ procesu,N”.
`log_level_decrease[<target>]`	Zmniejsz poziom rejestrowania. Jeśli element docelowy nie zostanie określony, dotyczy to wszystkich procesów.	
`userparameter_reload`	Ponownie wczytuje wartości opcji `UserParameter` i `Include` z bieżącego pliku konfiguracyjnego.	

Przykłady:

- Zwiększenie poziomu rejestrowania wszystkich procesów

- Zwiększenie poziomu rejestrowania trzeciego procesu listener
- Zwiększenie poziomu rejestrowania procesu o PID 1234
- Zmniejszenie poziomu rejestrowania wszystkich procesów kontroli aktywnej

```
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:
Sterowanie w czasie działania nie jest obsługiwane w systemach OpenBSD, NetBSD i Windows.

Typy procesów agenta Typy procesów agenta są następujące:

- `active checks` - proces do wykonywania aktywnych kontroli;
- `collector` - proces do zbierania danych;
- `listener` - proces do nasłuchiwania pasywnych kontroli.

Plik dziennika agenta może służyć do obserwowania tych typów procesów.

Plik dziennika agenta jest tworzony z uprawnieniami do odczytu i zapisu wyłącznie dla właściciela pliku. Dodatkowo plik jest czytelny dla grupy właściciela. Wszystkie pozostałe uprawnienia są odrzucone.

Uruchamianie agenta jako użytkownik agent Zabbix w systemie UNIX jest przeznaczony do uruchamiania jako użytkownik inny niż root. Będzie działać jako dowolny użytkownik inny niż root, z którego uprawnieniami został uruchomiony. Dlatego agent można uruchomić jako dowolnego użytkownika innego niż root bez żadnych problemów.

Jeśli spróbujesz uruchomić go jako „root", przełączy się na zakodowanego na stałe użytkownika „zabbix", który musi istnieć w systemie. agenta można uruchomić jako „root" tylko po odpowiedniej modyfikacji parametru `AllowRoot` w pliku konfiguracyjnym agenta.

Plik konfiguracyjny Domyślna ścieżka do pliku konfiguracyjnego zależy od systemu operacyjnego i metody instalacji Zabbix:

- W systemie UNIX domyślna ścieżka to:
  - W przypadku instalacji ze źródeł: `/usr/local/etc/zabbix_agentd.conf` lub ścieżka określona przez zmienne w czasie kompilacji `--sysconfdir` lub `--prefix`.
  - W przypadku instalacji z pakietów i urządzenia: `/etc/zabbix/zabbix_agentd.conf`.
- W systemie Windows domyślna ścieżka to `C:\Program Files\Zabbix Agent\zabbix_agentd.conf`.

Jeśli na hoście działa pojedyncza instancja agenta, może ona używać domyślnego pliku konfiguracyjnego lub pliku konfiguracyjnego określonego w wierszu poleceń. W przypadku wielu instancji każda instancja agenta musi mieć własny plik konfiguracyjny (jedna z instancji może używać domyślnego pliku konfiguracyjnego).

Szczegółowe informacje dotyczące konfigurowania agenta Zabbix zawierają opcje pliku konfiguracyjnego dla agenta Zabbix w systemie Unix lub agenta Zabbix w systemie Windows.

Lokalizacja Należy pamiętać, że agent wymaga lokalizacji UTF-8, aby niektóre tekstowe pozycje agenta mogły zwracać oczekiwaną zawartość. Większość nowoczesnych systemów typu Unix ma domyślnie ustawioną lokalizację UTF-8, jednak w niektórych systemach może być konieczne jej ręczne ustawienie.

Kod wyjścia Agent Zabbix zwraca 0 w przypadku pomyślnego zakończenia i 1 w przypadku niepowodzenia.

Protokoły pasywnych i aktywnych sprawdzeń

Ta sekcja zawiera szczegóły protokołu dotyczące pasywnych i aktywnych kontroli wykonywanych przez Zabbix agent i Zabbix agent 2.

Zabbix używa protokołu komunikacyjnego opartego na JSON do komunikacji z agentami.

Protokoły Zabbix agent i Zabbix agent 2 zostały ujednolicone od Zabbix 7.0. Różnica między żądaniami i odpowiedziami Zabbix agent oraz Zabbix agent 2 jest wyrażona przez wartość znacznika "variant".

Pasywne sprawdzenia

Pasywne sprawdzenie to proste żądanie danych. Serwer Zabbix lub proxy pyta o określone dane (na przykład obciążenie CPU), a agent Zabbix odsyła wynik do serwera.

Pasywne sprawdzenia są wykonywane asynchronicznie - nie jest wymagane otrzymanie odpowiedzi na jedno żądanie przed rozpoczęciem innych sprawdzeń. Rozwiązywanie DNS również odbywa się asynchronicznie.

Agent poller będzie próbował połączyć się ze wszystkimi adresami zwróconymi przez wyszukiwanie DNS. Zapewnia to, że jeśli jeden adres IP jest niedostępny, poller spróbuje następnego dostępnego adresu, zwiększając prawdopodobieństwo pomyślnego połączenia. To ulepszenie dotyczy zarówno serwera Zabbix, jak i proxy.

Maksymalna współbieżność asynchronicznych sprawdzeń wynosi 1000 (zdefiniowana przez parametr MaxConcurrentChecksPerPoller).

Liczba asynchronicznych agent pollers jest definiowana przez parametr StartAgentPollers.

Żądanie serwera

Definicję nagłówka i długości danych można znaleźć w sekcji szczegóły protokołu.

```
{
  "request": "passive checks",
  "data": [
    {
      "key": "agent.version",
      "timeout": 3
    }
  ]
}
```

Field		Type	Mandatory	Value
request		*string*	yes	"passive checks"
data		*array of object*	yes	Element sprawdzenia pasywnego.
	key	*string*	yes	Klucz pozycji z rozwiniętymi makrami.
	timeout	*number*	yes	Limit czasu komunikacji.

Odpowiedź agenta

```
{
  "version": "8.0.0",
  "variant": 2,
  "data": [
    {
      "value": "8.0.0"
    }
  ]
}
```

Field		Type	Mandatory	Value
version		*string*	yes	Numer wersji agenta.
variant		*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).
data		*array of object*	yes	Zawiera wynik sprawdzenia.
	value	*string*	no	Wartość pozycji, jeśli sprawdzenie zakończyło się powodzeniem.
	error	*string*	no	Komunikat błędu, jeśli sprawdzenie nie zakończyło się powodzeniem.

Na przykład, dla obsługiwanych pozycji:

1. Serwer otwiera połączenie TCP
2. Serwer wysyła **<HEADER><DATALEN>{”request”:”passive checks”,”data”:[{”key”:”agent.ping”,”timeout”:3}]}**
3. Agent odczytuje żądanie i odpowiada **<HEADER><DATALEN>{”version”:”8.0.0”,”variant”:2,”data”:[{”value”:1}]}**
4. Serwer przetwarza dane, aby uzyskać wartość, w naszym przypadku ’1’
5. Połączenie TCP zostaje zamknięte

Dla nieobsługiwanych pozycji:

1. Serwer otwiera połączenie TCP
2. Serwer wysyła **<HEADER><DATALEN>{”request”:”passive checks”,”data”:[{”key”:”vfs.fs.size[/nono]”,”timeout”:3}]}**

3. Agent odczytuje żądanie i odpowiada **<HEADER><DATALEN>{”version”:”8.0.0”,”variant”:2,”data”:[{”error”:”Unsupported item key.”}]}**
4. Serwer przetwarza dane, zmienia stan pozycji na nieobsługiwany z określonym komunikatem błędu
5. Połączenie TCP zostaje zamknięte

Przełączenie awaryjne na stary protokół

Aby upewnić się, że serwer lub proxy Zabbixa może współpracować z agentami z wersji wcześniejszych niż 7.2, które używają protokołu tekstowego, zaimplementowano przełączenie awaryjne na stary protokół.

Kontrole pasywne są wykonywane przy użyciu protokołu JSON (7.0 i nowsze) po ponownym uruchomieniu lub po zmianie konfiguracji interfejsu.
Jeśli w odpowiedzi nie zostanie odebrany prawidłowy JSON (agent wysłał ”ZBX_NOTSUPPORTED”), Zabbix zapisze interfejs jako używający starego protokołu i **ponowi** próbę sprawdzenia, wysyłając tylko klucz pozycji.

Należy pamiętać, że co godzinę serwer/proxy Zabbixa ponownie spróbuje pracować z nowym protokołem ze wszystkimi interfejsami, w razie potrzeby przechodząc z powrotem na stary protokół.

Aktywne sprawdzanie

Aktywne sprawdzanie wymaga bardziej złożonego przetwarzania. agent musi najpierw pobrać z serwera/proxy listę pozycji i/lub poleceń zdalnych do niezależnego przetwarzania.

serwery/proxy, z których pobierane są aktywne sprawdzania, są wymienione w parametrze `ServerActive` pliku konfiguracyjnego agent. Częstotliwość żądania tych sprawdzań określa parametr `RefreshActiveChecks` w tym samym pliku konfiguracyjnym. Jeśli jednak odświeżenie aktywnych sprawdzań zakończy się niepowodzeniem, zostanie ponowione po zakodowanych na stałe 60 sekundach.

Note:
Od Zabbix 6.4 agent (w trybie aktywnym) nie otrzymuje już od serwera/proxy pełnej kopii konfiguracji co dwie minuty (domyślnie). Zamiast tego, aby zmniejszyć ruch sieciowy i zużycie zasobów, co 5 sekund (domyślnie) wykonywana jest przyrostowa synchronizacja konfiguracji, podczas której serwer/proxy udostępnia pełną kopię konfiguracji **tylko** wtedy, gdy agent jeszcze jej nie otrzymał albo gdy coś zmieniło się w konfiguracji hosta, globalnych makrach lub globalnych wyrażeniach regularnych.

Agent następnie okresowo wysyła nowe wartości do serwera(ów). Jeśli agent otrzymał jakiekolwiek zdalne polecenia do wykonania, wynik wykonania również zostanie wysłany. Należy pamiętać, że wykonywanie zdalnych poleceń na aktywnym agencie jest obsługiwane od Zabbix agent 7.0.

Note:
Jeśli agent znajduje się za zaporą sieciową, możesz rozważyć używanie wyłącznie aktywnych kontroli, ponieważ w takim przypadku nie trzeba modyfikować zapory, aby zezwolić na początkowe połączenia przychodzące.

Pobieranie listy pozycji

Żądanie agenta

Żądanie aktywnych kontroli służy do pobrania aktywnych kontroli, które mają zostać przetworzone przez agent.
To żądanie jest wysyłane przez agent po uruchomieniu, a następnie w odstępach 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	Nazwa hosta.
host_metadata	*string*	no	Wartość parametru konfiguracyjnego HostMetadata lub metryki HostMetadataItem.

Field	Type	Mandatory	Value
interface	*string*	no	Wartość parametru konfiguracyjnego HostInterface lub metryki HostInterfaceItem.
ip	*string*	no	Pierwszy adres IP parametru konfiguracyjnego ListenIP, jeśli jest ustawiony.
port	*number*	no	Wartość parametru konfiguracyjnego ListenPort, jeśli jest ustawiona i nie jest domyślnym portem nasłuchu agenta.
version	*string*	yes	Numer wersji agenta.
variant	*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).
config_revision	*number*	no	Identyfikator konfiguracji dla inkrementalnej synchronizacji konfiguracji.
session	*string*	no	Identyfikator sesji dla inkrementalnej synchronizacji konfiguracji.

Odpowiedź serwera

Odpowiedź aktywnych kontroli jest wysyłana przez serwer z powrotem do agent po przetworzeniu żądania aktywnych kontroli.

```
{
  "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	Informacje o błędzie w przypadku niepowodzenia.
data	*array of objects*	no	Pozycje aktywnych kontroli. Pomijane, jeśli konfiguracja hosta nie uległa zmianie.
key	*string*	no	Klucz pozycji z rozwiniętymi makrami.
itemid	*number*	no	Identyfikator pozycji.
delay	*string*	no	Interwał aktualizacji pozycji. Elastyczne/interwały harmonogramu są obsługiwane przez Zabbix agent i Zabbix agent 2 od Zabbix 7.0.
lastlogsize	*number*	no	lastlogsize pozycji.
mtime	*number*	no	mtime pozycji.
timeout	*string*	no	Limit czasu pozycji.
refresh_unsupported	*number*	no	Interwał odświeżania nieobsługiwanej pozycji.
regexp	*array of objects*	no	Globalne wyrażenia regularne.
name	*string*	no	Nazwa globalnego wyrażenia regularnego.

Field	Type	Mandatory	Value
expression	*string*	no	Globalne wyrażenie regularne.
expression_type	*number*	no	Typ globalnego wyrażenia regularnego.
exp_delimiter	*string*	no	Separator globalnego wyrażenia regularnego.
case_sensitive	*number*	no	Ustawienie rozróżniania wielkości liter dla globalnego wyrażenia regularnego.
commands	*array of objects*	no	Zdalne polecenia do wykonania. Dołączane, jeśli wykonanie zdalnego polecenia zostało wyzwolone przez operację action lub ręczne wykonanie script. Należy pamiętać, że wykonywanie zdalnych poleceń na aktywnym agencie jest obsługiwane od Zabbix agent 7.0. Starsze aktywne agenty zignorują wszystkie zdalne polecenia zawarte w odpowiedzi serwera aktywnych kontroli.
command	*string*	no	Zdalne polecenie.
id	*number*	no	Identyfikator zdalnego polecenia.
wait	*number*	no	Tryb wykonania zdalnego polecenia (”0” (nowait) dla poleceń z operacji action; ”1” (wait) dla poleceń z ręcznego wykonania script).
timeout	*number*	no	Limit czasu wykonania zdalnego polecenia w konfiguracji serwer/proxy.
config_revision	*number*	no	Identyfikator konfiguracji dla inkrementalnej synchronizacji konfiguracji. Pomijane, jeśli konfiguracja hosta nie uległa zmianie. Zwiększane, jeśli konfiguracja hosta została zmieniona.

Serwer musi odpowiedzieć sukcesem.

Na przykład:

1. Agent otwiera połączenie TCP
2. Agent prosi o listę kontroli
3. Serwer odpowiada listą pozycji i zdalnych poleceń do wykonania
4. Agent analizuje odpowiedź
5. Połączenie TCP zostaje zamknięte
6. Agent rozpoczyna okresowe zbieranie danych i wykonuje zdalne polecenia (obsługiwane od Zabbix agent 7.0)

Attention:
Należy pamiętać, że (wrażliwe) dane konfiguracyjne mogą stać się dostępne dla podmiotów mających dostęp do portu trapper serwera Zabbix podczas korzystania z aktywnej kontroli. Jest to możliwe, ponieważ każdy może podszyć się pod aktywnego agenta i zażądać danych konfiguracyjnych pozycji; uwierzytelnianie nie jest przeprowadzane, chyba że użyjesz opcji encryption.

Wysyłanie zebranych danych

Agent wysyła

Żądanie danych agenta zawiera zebrane wartości pozycji oraz wartości wykonanych zdalnych poleceń (jeśli występują).

```
{
  "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	Wartości pozycji.
id	*number*	yes	Identyfikator wartości (inkrementowany licznik używany do sprawdzania zduplikowanych wartości w przypadku problemów sieciowych).
itemid	*number*	yes	Identyfikator pozycji.
value	*string*	no	Wartość pozycji.
lastlogsize	*number*	no	lastlogsize pozycji.
mtime	*number*	no	mtime pozycji.
state	*number*	no	Stan pozycji.
source	*string*	no	Źródło wpisu dziennika zdarzeń wartości.
eventid	*number*	no	eventid wpisu dziennika zdarzeń wartości.
severity	*number*	no	severity wpisu dziennika zdarzeń wartości.
timestamp	*number*	no	timestamp wpisu dziennika zdarzeń wartości.
clock	*number*	yes	Znacznik czasu wartości (sekundy od epoki).
ns	*number*	yes	Nanosekundy znacznika czasu wartości.
commands	*array of objects*	no	Wynik wykonania zdalnych poleceń. Należy pamiętać, że wykonywanie zdalnych poleceń na aktywnym agencie jest obsługiwane od Zabbix agent 7.0. Starsze aktywne agenty zignorują wszystkie zdalne polecenia zawarte w odpowiedzi serwera na aktywne sprawdzenia.
id	*number*	no	Identyfikator zdalnego polecenia.
value	*string*	no	Wynik wykonania zdalnego polecenia, jeśli wykonanie zakończyło się powodzeniem.
error	*string*	no	Komunikat błędu wykonania zdalnego polecenia, jeśli wykonanie nie powiodło się.
session	*string*	yes	Unikalny identyfikator sesji generowany za każdym razem po uruchomieniu agenta (32 cyfry szesnastkowe).
host	*string*	yes	Nazwa hosta.
version	*string*	yes	Numer wersji agenta.
variant	*number*	yes	Wariant agenta (*1* - Zabbix agent, *2* - Zabbix agent 2).

Każdej wartości przypisywany jest wirtualny identyfikator. Identyfikator wartości jest prostym, rosnącym licznikiem, unikalnym w ramach jednej sesji danych (identyfikowanej przez token sesji). Ten identyfikator służy do odrzucania zduplikowanych wartości, które mogą być wysyłane w środowiskach o słabej łączności.

Odpowiedź serwera

Odpowiedź na dane agenta jest wysyłana przez serwer z powrotem do agenta po przetworzeniu żądania danych agenta.

```
{
  "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	Wyniki przetwarzania pozycji.

Attention:
Jeśli wysyłanie niektórych wartości nie powiedzie się na serwerze (na przykład dlatego, że host lub pozycja zostały wyłączone albo usunięte), agent nie będzie ponawiał wysyłania tych wartości.

Na przykład:

1. Agent otwiera połączenie TCP
2. Agent wysyła listę wartości
3. Serwer przetwarza dane i odsyła status
4. Połączenie TCP zostaje zamknięte

Attention:
Komunikat błędu zostanie po stronie serwera skrócony do 2048 znaków.

Komunikat heartbeat

Wysyłany przez agenta

Komunikat heartbeat jest wysyłany przez aktywnego agenta do serwera/proxy Zabbix co `HeartbeatFrequency` sekund (konfigurowane w pliku konfiguracyjnym agenta Zabbix lub agenta Zabbix 2).

Służy do monitorowania dostępności aktywnych kontroli.

```
{
  "request": "active check heartbeat",
  "host": "Zabbix server",
  "heartbeat_freq": 60,
  "version": "8.0.0",
  "variant": 2
}
```

Pole	Typ	Wymagane	Wartość
request	*string*	tak	`active check heartbeat`
host	*string*	tak	Nazwa hosta.
heartbeat_freq	*number*	tak	Częstotliwość komunikatów heartbeat agenta (parametr konfiguracyjny HeartbeatFrequency).
version	*string*	tak	Numer wersji agenta.
variant	*number*	tak	Wariant agenta (*1* - agent Zabbix, *2* - agent Zabbix 2).

Odpowiedź przekierowania

Gdy host został ponownie przypisany, serwer może nakazać agentowi przekierowanie komunikatów heartbeat (oraz kolejnych aktywnych kontroli) do innego proxy lub wystąpienia serwera.

```
  {
    "response": "failed",
    "redirect": {
      "revision": 2,
      "address": "192.0.2.1:10055"
    }
  }
```

Pole		Typ	Wymagane	Wartość
response		*string*	tak	`success` | `failed`
redirect		*object*	tak	Instrukcje przekierowania.
	revision	*number*	tak	Identyfikator rewizji konfiguracji.
	address	*string*	tak	Adres docelowego serwera/proxy.

Starszy protokół XML

Note:
Zabbix przyjmie do 16 MB danych XML zakodowanych w formacie Base64, ale pojedyncza zdekodowana wartość nie powinna być większa niż 64 KB; w przeciwnym razie podczas dekodowania zostanie obcięta do 64 KB.

Uruchamianie agenta jako root

Od Zabbix **5.0.0** plik usługi systemd dla agenta Zabbix w oficjalnych pakietach jawnie zawiera dyrektywy `User` i `Group`. Obie są ustawione na `zabbix`.

Nie jest już możliwe skonfigurowanie, jako który użytkownik ma działać agent Zabbix, za pomocą pliku `zabbix_agentd.conf`, ponieważ agent ominie tę konfigurację i uruchomi się jako użytkownik określony w pliku usługi systemd. Aby uruchomić agenta Zabbix jako root, należy wprowadzić zmiany opisane poniżej.

Zabbix agent

Aby zastąpić domyślnego użytkownika i grupę dla Zabbix agent, uruchom:

```
systemctl edit zabbix-agent
```

Następnie dodaj następującą zawartość:

```
[Service]
User=root
Group=root
```

Przeładuj demony i uruchom ponownie usługę zabbix-agent:

```
systemctl daemon-reload
systemctl restart zabbix-agent
```

W przypadku **Zabbix agent** ponownie włącza to funkcję konfigurowania użytkownika w pliku konfiguracyjnym agenta. Teraz należy ustawić parametry konfiguracyjne `User=root` i `AllowRoot=1` w pliku konfiguracyjnym.

Zabbix agent 2

Aby zastąpić domyślnego użytkownika i grupę dla Zabbix agent 2, uruchom:

```
systemctl edit zabbix-agent2
```

Następnie dodaj następującą zawartość:

```
[Service]
User=root
Group=root
```

Załaduj ponownie demony i uruchom ponownie usługę zabbix-agent2:

```
systemctl daemon-reload
systemctl restart zabbix-agent2
```

W przypadku **Zabbix agent2** całkowicie określa to użytkownika, jako którego działa. Nie są wymagane żadne dodatkowe modyfikacje.

Parametry konfiguracji: Zabbix agent

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania agent.

Wartości parametrów należy określić w pliku konfiguracji agent Zabbix (`zabbix_agentd.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
Alias	Ustawia alias dla klucza pozycji.
AllowKey	Zezwala na wykonywanie tych kluczy pozycji, które pasują do wzorca.
AllowKeyRegexp	Zezwala na wykonywanie kluczy pozycji, które pasują do wzorca opartego na wyrażeniu regularnym.
AllowRoot	Zezwala agent na uruchamianie jako `root`.
BufferSend	Ustawia częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix.
BufferSize	Ustawia maksymalną liczbę wartości w buforze pamięci.
DebugLevel	Ustawia poziom debugowania.

Parametr	Opis
DenyKey	Zabrania wykonywania tych kluczy pozycji, które pasują do wzorca.
DenyKeyRegexp	Zabrania wykonywania kluczy pozycji, które pasują do wzorca opartego na wyrażeniu regularnym.
EnableRemoteCommands	Zezwala na zdalne polecenia z serwera Zabbix. Ten parametr jest przestarzały; zamiast niego użyj `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.
HeartbeatFrequency	Ustawia częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Określa interfejs host.
HostInterfaceItem	Określa pozycję służącą do pobierania interfejsu host.
HostMetadata	Określa metadane host.
HostMetadataItem	Określa pozycję agent Zabbix służącą do pobierania metadanych host.
Hostname	Określa nazwę host.
HostnameItem	Określa pozycję agent Zabbix służącą do pobierania nazwy host.
Include	Określa katalog lub pojedyncze pliki, które mają zostać uwzględnione w pliku konfiguracji.
ListenBacklog	Ustawia maksymalną liczbę oczekujących połączeń w kolejce TCP.
ListenIP	Określa adres IP nasłuchiwania agent dla połączeń z serwera/proxy.
ListenPort	Określa port nasłuchiwania agent.
LoadModule	Określa moduł ładowany podczas uruchamiania agent.
LoadModulePath	Określa pełną ścieżkę do lokalizacji modułów agent.
LogFile	Określa plik dziennika agent.
LogFileSize	Ustawia maksymalny rozmiar pliku dziennika.
LogRemoteCommands	Włącza rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
LogType	Określa typ wyjścia dziennika.
MaxLinesPerSecond	Ustawia maksymalną liczbę nowych wierszy na sekundę wysyłanych do serwera/proxy Zabbix podczas przetwarzania aktywnych kontroli `log` lub `logrt`.
PidFile	Określa plik PID.
RefreshActiveChecks	Określa częstotliwość odświeżania listy aktywnych kontroli.
Server	Określa adresy IP lub nazwy DNS serwerów/proxy Zabbix.
ServerActive	Określa adres serwera/proxy Zabbix lub konfigurację klastra, z którego mają być pobierane aktywne kontrole.
SourceIP	Określa źródłowy adres IP.
StartAgents	Ustawia liczbę wstępnie rozgałęzionych instancji `zabbix_agentd` do przetwarzania pasywnych kontroli.
Timeout	Określa, jak długo (w sekundach) należy czekać na ustanowienie połączenia i wymianę danych z proxy lub serwerem Zabbix.
TLSAccept	Określa wymagany poziom szyfrowania dla połączeń przychodzących.
TLSCAFile	Określa pełną nazwę ścieżki pliku zawierającego certyfikaty najwyższego poziomu CA do weryfikacji certyfikatu peera, używane w szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Określa pełną nazwę ścieżki pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używane w szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Określa ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Określa ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Określa ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Określa ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Określa ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Określa ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Określa wymagany poziom szyfrowania dla połączeń z serwerem lub proxy Zabbix.
TLSCRLFile	Określa pełną nazwę ścieżki pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Określa pełną nazwę ścieżki pliku zawierającego klucz prywatny agent, używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Określa pełną nazwę ścieżki pliku zawierającego klucz wstępnie współdzielony agent, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSPSKIdentity	Określa ciąg tożsamości klucza wstępnie współdzielonego, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSServerCertIssuer	Określa dozwolonego wystawcę certyfikatu serwera (proxy).

Parametr	Opis
TLSServerCertSubject	Określa dozwolony podmiot certyfikatu serwera (proxy).
UnsafeUserParameters	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
User	Obniża uprawnienia do określonego, istniejącego użytkownika w systemie.
UserParameter	Określa parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Określa domyślną ścieżkę wyszukiwania poleceń UserParameter.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne demona, a nie wartości w dostarczonych plikach konfiguracji.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracji wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Alias

Ustaw alias dla klucza pozycji. Można go użyć do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów `Alias`. Nie można używać wielu parametrów z takim samym kluczem `Alias`.
 Różne klucze `Alias` mogą odwoływać się do tego samego klucza pozycji.
 Aliasów można używać w parametrze `HostMetadataItem`, ale nie w parametrze `HostnameItem`.

Przykład 1: Pobieranie identyfikatora użytkownika `zabbix`.

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Teraz można użyć skróconego klucza `zabbix.userid` do pobierania danych.

Przykład 2: Pobieranie wykorzystania procesora z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Umożliwia to użycie klucza `cpu.util` do pobierania procentowego wykorzystania procesora z parametrami domyślnymi, a także użycie `cpu.util[all, idle, avg15]` do pobierania określonych danych dotyczących wykorzystania procesora.

Przykład 3: Uruchamianie wielu reguł niskopoziomowego wykrywania przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz można skonfigurować kilka reguł wykrywania przy użyciu `vfs.fs.discovery`, z różnymi parametrami dla każdej reguły, np. `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym, które obsługuje znak * do dopasowania dowolnej liczby dowolnych znaków.
 Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z `DenyKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Kolejność określa, która z pasujących reguł będzie miała priorytet. Zobacz także: Ograniczanie sprawdzeń agent.

AllowKeyRegexp

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `DenyKeyRegexp`, `AllowKey` i `DenyKey`. Parametry są przetwarzane jeden po drugim, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie kontroli agent.

Przykład:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

AllowRoot

Zezwalaj agentowi na uruchamianie się jako `root`. Jeśli ta opcja jest wyłączona, a agent zostanie uruchomiony przez użytkownika `root`, agent spróbuje zamiast tego przełączyć się na użytkownika `zabbix`. Nie ma to żadnego wpływu, jeśli agent został uruchomiony przez zwykłego użytkownika.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj; 1 - zezwalaj

BufferSend

Ustaw częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix (w sekundach).

Domyślnie: 5
 Zakres: 1-3600

BufferSize

Ustaw maksymalną liczbę wartości w buforze pamięci.
 Agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor się zapełni.

Domyślnie: 100
 Zakres: 2-65535

DebugLevel

Ustaw poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak *, który dopasowuje dowolną liczbę dowolnych znaków.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `AllowKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie kontroli agent.

Domyślnie: `DenyKey=system.run[*]`

DenyKeyRegexp

Odmów wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można zdefiniować wiele reguł dopasowywania kluczy w połączeniu z `AllowKeyRegexp`, `AllowKey` i `DenyKey`. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo.
 Wszystkie pozycje `system.run[]` są domyślnie odrzucane (`DenyKey=system.run[*]`), chyba że określono inną regułę `system.run[*]`.

Zobacz także: Ograniczanie kontroli agent.

EnableRemoteCommands

Zezwalaj na zdalne polecenia z serwera Zabbix.
 Ten parametr jest **przestarzały**. Zamiast niego użyj `AllowKey=system.run[*]`, `AllowKeyRegexp=^system\.run\[.*\]$`, `DenyKey=system.run[*]` lub `DenyKeyRegexp=^system\.run\[.*\]$`.
 Jest to wewnętrzny alias parametrów `AllowKey`, `AllowKeyRegexp`, `DenyKey` i `DenyKeyRegexp`, zależny od wartości: - 0 - `DenyKey=system.run[*]` (`DenyKeyRegexp=^system\.run\[.*\]$`) - 1 - `AllowKey=system.run[*]` (`AllowKeyRegexp=^system\.`

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

HeartbeatFrequency

Ustaw częstotliwość wiadomości heartbeat w sekundach. Używane do monitorowania dostępności aktywnych kontroli.
0 - wiadomości heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Określ interfejs hosta (adres IP lub nazwa DNS) używany podczas autorejestracji hosta. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawne skonfigurowanie adresu IP lub DNS. Więcej informacji znajdziesz w sekcji Using DNS as default interface.

Jeśli nie zostanie zdefiniowana, wartość zostanie pobrana z `HostInterfaceItem`.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Określ pozycję służącą do ustalenia interfejsu hosta (adresu IP lub nazwy DNS) podczas automatycznej rejestracji hosta. Ta wartość jest używana tylko wtedy, gdy parametr `HostInterface` nie został zdefiniowany. Więcej informacji zawiera sekcja Używanie DNS jako domyślnego interfejsu.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey, AllowKeyRegexp, DenyKey lub DenyKeyRegexp.

HostMetadata

Określ metadane służące do identyfikowania lub rozróżniania hosta podczas automatycznej rejestracji hosta (aktywny agent). HostMetadata umożliwia rozróżnianie hostów niezależnie od ich nazw.

Jeśli parametr nie zostanie zdefiniowany, wartość zostanie pobrana z HostMetadataItem.

agent zgłosi błąd i nie uruchomi się, jeśli określona wartość przekroczy limit 2034 bajtów lub będzie ciągiem znaków innym niż UTF-8.
Jeśli parametr wymaga adresu IP lub nazwy DNS, wartości poprawne w UTF-8, ale niebędące prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Zakres: 0–2034 bajtów

HostMetadataItem

Określa pozycję agenta Zabbix, z której mają być pobierane metadane hosta. Ta opcja jest używana tylko wtedy, gdy parametr HostMetadata nie został zdefiniowany.

Wartość HostMetadataItem jest pobierana przy każdej próbie automatycznej rejestracji i używana wyłącznie podczas procesu automatycznej rejestracji hosta (aktywny agent). HostMetadataItem umożliwia rozróżnianie hostów na podstawie innych informacji niż nazwa hosta.

Obsługiwane są parametry użytkownika i aliasy. Pozycja system.run[] jest obsługiwana niezależnie od ustawień AllowKey, AllowKeyRegexp, DenyKey lub DenyKeyRegexp.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy w dzienniku, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem UTF-8; w przeciwnym razie zostanie zignorowana. Jeśli parametr oczekuje adresu IP lub nazwy DNS, wartości, które są prawidłowe w UTF-8, ale nie są prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Nazwa hosta

Określ unikalne nazwy hostów z uwzględnieniem wielkości liter. Lista rozdzielona przecinkami.
 Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze.
 Wartość jest pobierana z HostnameItem, jeśli nie została zdefiniowana.
 Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Określ pozycję agenta Zabbix służącą do pobierania nazwy hosta.
 Ta opcja jest używana tylko wtedy, gdy parametr Hostname nie jest zdefiniowany.
 Parametry użytkownika ani aliasy nie są obsługiwane, ale pozycja system.run[] jest obsługiwana niezależnie od wartości AllowKey, AllowKeyRegexp, DenyKey lub DenyKeyRegexp.

Wartość domyślna: system.hostname

Include

Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracyjnego.
 Podczas instalacji Zabbix ten parametr jest ustawiany na /usr/local/etc, chyba że zmienisz to podczas kompilacji.

Ogólne zasady dla parametru Include:

- Ścieżka może być względna względem lokalizacji pliku zabbix_agentd.conf.
- W jednym pliku konfiguracyjnym można określić wiele parametrów Include.
- Zabbix przetwarza parametr Include natychmiast po jego odczytaniu, więc warto umieścić go na końcu pliku konfiguracyjnego.

Zasady dotyczące plików dołączanych:

- Każdy plik musi być możliwy do odczytu.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie uruchomi się.

Zasady dotyczące dołączania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności dołączania (np. pliki nie są dołączane alfabetycznie).
- Nie definiuj tego samego parametru w więcej niż jednym dołączonym pliku (np. aby zastąpić ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.

- Zwróć uwagę na kopie zapasowe, które niektóre edytory tekstu tworzą automatycznie (np. edycja `include/my_specific.conf` może utworzyć `include/my_specific.conf.BAK`), ponieważ oba pliki zostaną dołączone.

Aby dołączyć tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (`*`) do dopasowywania wzorca.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenBacklog

Ustaw maksymalną liczbę oczekujących połączeń w kolejce TCP.
 Wartość domyślna jest zakodowaną na stałe wartością, która zależy od systemu.
 Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać bez ostrzeżenia obcięte do „maksimum określonego przez implementację”.

Domyślnie: `SOMAXCONN`
 Zakres: 0 - INT_MAX

ListenIP

Określa adres IP, na którym agent nasłuchuje połączeń z serwera/proxy. Lista rozdzielana przecinkami.

Domyślnie: `0.0.0.0`

ListenPort

Określa port nasłuchu agent dla połączeń z serwera/proxy.

Domyślnie: 10050
 Zakres: 1024-32767

LoadModule

Określa moduł do załadowania podczas uruchamiania agent.
 Moduły służą do rozszerzania funkcjonalności agenta.
 Moduł musi znajdować się w katalogu określonym przez `LoadModulePath` albo ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od `/`), to `LoadModulePath` jest ignorowany.
 Formaty: - `LoadModule=<module.so>` - `LoadModule=<path/module.so>` - `LoadModule=</abs_path/module.so>`

Dozwolone jest podanie wielu parametrów `LoadModule`.

LoadModulePath

Określ pełną ścieżkę do lokalizacji modułów agenta.
 Domyślna wartość zależy od opcji kompilacji.

LogFile

Określ plik dziennika agent.

Obowiązkowe: Tak, jeśli `LogType` ma ustawioną wartość `file`; w przeciwnym razie nie

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
0 - wyłącza automatyczną rotację dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika i rotacja pliku z jakiegokolwiek powodu się nie powiedzie, istniejący plik dziennika zostanie obcięty i utworzony od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
 Polecenia będą rejestrowane tylko wtedy, gdy zostaną wykonane zdalnie.
 Wpisy w dzienniku nie będą tworzone, jeśli `system.run[]` zostanie uruchomione lokalnie przez parametry `HostMetadataItem`, `HostInterfaceItem` lub `HostnameItem`.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

LogType

Określ typ wyjścia logu: - *file* - zapisuj log do pliku określonego parametrem `LogFile`; - *system* - zapisuj log do syslog; - *console* - zapisuj log do standardowego wyjścia.

Domyślnie: `file`

MaxLinesPerSecond

Ustaw maksymalną liczbę nowych linii na sekundę wysyłanych do serwera Zabbix/proxy podczas przetwarzania aktywnych kontroli `log` lub `logrt`.
 Podana wartość zostanie zastąpiona przez parametr `maxlines`, określony w kluczu pozycja `log` lub `logrt`.
Uwaga: Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w `MaxLinesPerSecond`, aby wyszukać wymaganą frazę w pozycjach logów.

Domyślnie: 20
 Zakres: 1-1000

PidFile

Określa plik PID.

Domyślnie: `/tmp/zabbix_agentd.pid`

RefreshActiveChecks

Określa częstotliwość odświeżania listy aktywnych kontroli, w sekundach.
 Należy pamiętać, że po nieudanym odświeżeniu aktywnych kontroli następna próba odświeżenia zostanie podjęta po 60 sekundach.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Określ adresy IP, opcjonalnie w notacji CIDR, lub nazwy DNS serwerów/proxy Zabbixa. Lista rozdzielana przecinkami.
 Połączenia przychodzące będą akceptowane tylko z hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4. Należy pamiętać, że "IPv4-compatible IPv6 addresses" (`0000::/96 prefix`) są obsługiwane, ale przestarzałe zgodnie z RFC4291. Dozwolone są spacje.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak, jeśli StartAgents nie jest jawnie ustawione na 0

ServerActive

Określa adres serwera/proxy Zabbix lub konfigurację klastra, z którego mają być pobierane aktywne kontrole.
 Adres serwera/proxy to adres IP lub nazwa DNS oraz opcjonalny port oddzielony dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwerów lub proxy oddzielonych średnikiem. Można podać wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy podawać więcej niż jednego proxy Zabbix dla każdego serwera/klastra Zabbix. Jeśli podano proxy Zabbix, nie należy podawać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby używać kilku niezależnych serwerów Zabbix równolegle. Dozwolone są spacje.
Jeśli port nie zostanie określony, używany jest port domyślny.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla tego hosta określono port. Jeśli port nie jest określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne kontrole są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Określ adres IP źródłowy dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix;
- nawiązywania połączeń podczas wykonywania niektórych pozycji (`web.page.get`, `net.tcp.port` itp.).

StartAgents

Ustaw liczbę wcześniej utworzonych instancji `zabbix_agentd` do przetwarzania pasywnych kontroli.
 Jeśli ustawiono 0, pasywne kontrole są wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Domyślnie: 10
 Zakres: 0-100

Limit czasu

Określa, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr definiuje czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwanie na odpowiedź z serwera Zabbix;
- wysyłanie żądań do serwera Zabbix, w tym żądań konfiguracji pozycji i danych pozycji w aktywnych sprawdzeniach;
- pobieranie danych dziennika przez logfile;
- wysyłanie komunikatów heartbeat;
- maksymalny czas trwania sprawdzeń `vfs.*`;
- używanie przez moduły agenta Zabbix;
- działanie jako wartość awaryjna w scenariuszach, w których serwer lub proxy w wersji starszej niż 7.0 wysyła sprawdzenia bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych sprawdzeń agenta, które mają konfigurowalne ustawienia limitu czasu we frontendzie (na poziomie globalnym, proxy lub dla pojedynczej pozycji).

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Określ poziom szyfrowania wymagany dla połączeń przychodzących.
 Używane dla kontroli pasywnych.
 Można podać wiele wartości, oddzielając je przecinkiem: - *unencrypted* - akceptuj połączenia bez szyfrowania (domyślnie); - *psk* - akceptuj połączenia z TLS i kluczem wstępnie współdzielonym (PSK); - *cert* - akceptuj połączenia z TLS i certyfikatem.

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używaną do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używany do szyfrowanej komunikacji z komponentami Zabbix.

TLSCipherAll

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherAll13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherCert

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

TLSCipherCert13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

TLSCipherPSK

Określ ciąg priorytetów GnuTLS lub ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład dla GnuTLS:

```
NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIG
```

Przykład dla OpenSSL:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Określ ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego w TLS 1.3. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Określ poziom szyfrowania wymagany dla połączeń agent z serwerem Zabbix lub proxy.
 Używane dla aktywnych kontroli.
 Można określić tylko jedną wartość: - *unencrypted* - połączenie bez szyfrowania (domyślnie); - *psk* - połączenie z użyciem TLS i współdzielonego klucza wstępnego (PSK); - *cert* - połączenie z użyciem TLS i certyfikatu.

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz agenta, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Podaj pełną ścieżkę do pliku zawierającego współdzielony klucz agenta, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSPSKIdentity

Określ ciąg tożsamości pre-shared key, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSServerCertIssuer

Określ dozwolonego wystawcę certyfikatu serwera (proxy).

TLSServerCertSubject

Określ dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Dodatkowo znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

Użytkownik

Obniż uprawnienia do określonego, istniejącego użytkownika w systemie.
Ma to efekt tylko wtedy, gdy jest uruchamiane jako `root` i `AllowRoot` jest wyłączone.

Domyślnie: `zabbix`

UserParameter

Określ zdefiniowany przez użytkownika parametr do monitorowania. Może istnieć kilka zdefiniowanych przez użytkownika parametrów.
Format: `UserParameter=<key>,<shell command>`
Należy pamiętać, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie EOL. Polecenia powłoki mogą używać ścieżek względnych, jeśli określono parametr `UserParameterDir`.

Przykład:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Określa domyślną ścieżkę wyszukiwania dla poleceń `UserParameter`. Jeśli jest używana, agent zmieni swój katalog roboczy na ten wskazany tutaj przed wykonaniem polecenia. Dzięki temu polecenia `UserParameter` mogą używać względnego prefiksu `./` zamiast pełnej ścieżki.
Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=/opt/myscripts
```

Zobacz także

1. Różnice w konfiguracji Zabbix agent dla aktywnych i pasywnych kontroli od wersji 2.0.0

Parametry konfiguracji: Zabbix agent (Windows)

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania agent na systemie Windows.

Określ wartości parametrów w pliku konfiguracji agent Zabbix (`zabbix_agentd.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę).
- Szczegóły parametrów.

Parametr	Opis
Alias	Ustaw alias dla klucza pozycji.
AllowKey	Zezwól na wykonywanie kluczy pozycji pasujących do wzorca.
AllowKeyRegexp	Zezwól na wykonywanie kluczy pozycji pasujących do wzorca opartego na wyrażeniu regularnym.
BufferSend	Ustaw częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix.
BufferSize	Ustaw maksymalną liczbę wartości w buforze pamięci.
DebugLevel	Ustaw poziom debugowania.
DenyKey	Zabroń wykonywania kluczy pozycji pasujących do wzorca.
DenyKeyRegexp	Zabroń wykonywania kluczy pozycji pasujących do wzorca opartego na wyrażeniu regularnym.
EnableRemoteCommands	Zezwól na zdalne polecenia z serwera Zabbix. Ten parametr jest przestarzały; zamiast niego użyj `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.
HeartbeatFrequency	Ustaw częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Określ interfejs host.
HostInterfaceItem	Określ pozycję służącą do pobierania interfejsu host.
HostMetadata	Określ metadane host.
HostMetadataItem	Określ pozycję agent Zabbix służącą do pobierania metadanych host.
Hostname	Określ nazwę host.
HostnameItem	Określ pozycję agent Zabbix służącą do pobierania nazwy host.
Include	Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracji.
ListenBacklog	Ustaw maksymalną liczbę oczekujących połączeń w kolejce TCP.
ListenIP	Określ adres IP nasłuchiwania agent dla połączeń z serwera/proxy.
ListenPort	Określ port nasłuchiwania agent.
LogFile	Określ plik dziennika agent.
LogFileSize	Ustaw maksymalny rozmiar pliku dziennika.
LogRemoteCommands	Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
LogType	Określ typ wyjścia dziennika.
MaxLinesPerSecond	Ustaw maksymalną liczbę nowych wierszy na sekundę wysyłanych do serwera/proxy Zabbix podczas przetwarzania aktywnych kontroli `log`, `logrt` lub `eventlog`.
PerfCounter	Zdefiniuj nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach).
PerfCounterEn	Zdefiniuj nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach). W odróżnieniu od PerfCounter ścieżki liczników wydajności muszą być podane w języku angielskim.
RefreshActiveChecks	Określ częstotliwość odświeżania listy aktywnych kontroli.
Server	Określ adresy IP lub nazwy DNS serwerów/proxy Zabbix.
ServerActive	Określ adres serwera/proxy Zabbix lub konfigurację klastra, z którego mają być pobierane aktywne kontrole.
SourceIP	Określ źródłowy adres IP.

Parametr	Opis
StartAgents	Ustaw liczbę wstępnie rozgałęzionych instancji `zabbix_agentd` do przetwarzania pasywnych kontroli.
Timeout	Określ czas oczekiwania (w sekundach) na ustanowienie połączenia i wymianę danych z serwerem lub proxy Zabbix.
TLSAccept	Określ wymagany poziom szyfrowania dla połączeń przychodzących.
TLSCAFile	Określ pełną ścieżkę do pliku zawierającego certyfikaty głównego urzędu certyfikacji (CA) do weryfikacji certyfikatu peera, używane w szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Określ pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używane w szyfrowanej komunikacji między komponentami Zabbix.
TLSConnect	Określ wymagany poziom szyfrowania dla połączeń z serwerem lub proxy Zabbix.
TLSCRLFile	Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Określ pełną ścieżkę do pliku zawierającego klucz prywatny agent, używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Określ pełną ścieżkę do pliku zawierającego klucz wstępnie współdzielony agent, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSPSKIdentity	Określ ciąg identyfikatora klucza wstępnie współdzielonego, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSServerCertIssuer	Określ dozwolonego wystawcę certyfikatu serwera (proxy).
TLSServerCertSubject	Określ dozwolony podmiot certyfikatu serwera (proxy).
UnsafeUserParameters	Zezwól na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
UserParameter	Określ parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Określ domyślną ścieżkę wyszukiwania poleceń UserParameter.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne demona, a nie wartości zawarte w dostarczanych plikach konfiguracji.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracji wyłącznie w kodowaniu UTF-8, bez BOM.
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Alias

Ustaw alias dla klucza pozycji. Można go użyć do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów `Alias`. Nie można używać wielu parametrów z takim samym kluczem `Alias`.
 Różne klucze `Alias` mogą odwoływać się do tego samego klucza pozycji.
 Aliasów można używać w `HostMetadataItem`, ale nie w parametrach `HostnameItem` ani `PerfCounter`.

Przykład 1: Pobieranie procentowego użycia pliku stronicowania z serwera.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Teraz można użyć skróconego klucza `pg_usage` do pobierania danych.

Przykład 2: Pobieranie obciążenia procesora z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Umożliwia to użycie klucza `cpu.load` do pobierania obciążenia procesora z parametrami domyślnymi, a także użycie `cpu.load[percpu,avg15]` do pobierania szczegółowych danych dotyczących obciążenia procesora.

Przykład 3: Uruchamianie wielu reguł niskopoziomowego wykrywania przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz można skonfigurować kilka reguł wykrywania przy użyciu `vfs.fs.discovery`, z różnymi parametrami dla każdej reguły, np. `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym, które obsługuje znak * do dopasowania dowolnej liczby dowolnych znaków.
 Można zdefiniować wiele reguł dopasowania kluczy w połączeniu z `DenyKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością

ich występowania. Kolejność określa, która z pasujących reguł będzie miała priorytet. Zobacz także: Ograniczanie sprawdzeń agenta.

AllowKeyRegexp

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można zdefiniować wiele reguł dopasowywania kluczy w połączeniu z DenyKeyRegexp, AllowKey i DenyKey. Parametry są przetwarzane jeden po drugim, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie sprawdzania przez agent.

Przykład:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

BufferSend

Ustaw częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix (w sekundach).

Domyślnie: 5
 Zakres: 1-3600

BufferSize

Ustaw maksymalną liczbę wartości w buforze pamięci.
 Agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor zostanie zapełniony.

Domyślnie: 100
 Zakres: 2-65535

DebugLevel

Ustaw poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak *, który dopasowuje dowolną liczbę dowolnych znaków.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z AllowKey, AllowKeyRegexp i DenyKeyRegexp. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie sprawdzania przez agent.

Wartość domyślna: DenyKey=system.run[*]

DenyKeyRegexp

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można zdefiniować wiele reguł dopasowywania kluczy w połączeniu z AllowKeyRegexp, AllowKey i DenyKey. Parametry są przetwarzane po kolei, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo.
 Wszystkie pozycje system.run[] są domyślnie zabronione (DenyKey=system.run[*]), chyba że określono inną regułę system.run[*].

Zobacz także: Ograniczanie kontroli agentów.

EnableRemoteCommands

Zezwalaj na zdalne polecenia z serwera Zabbix.
 Ten parametr jest **przestarzały**. Zamiast niego użyj AllowKey=system.run[*], AllowKeyRegexp=^system\.run\[.*\]$, DenyKey=system.run[*] lub DenyKeyRegexp=^system\.run\[.*\]$.
 Jest to wewnętrzny alias parametrów AllowKey, AllowKeyRegexp, DenyKey i DenyKeyRegexp, zależny od wartości: - 0 - DenyKey=system.run[*] (DenyKeyRegexp=^system\.run\[.*\]$) - 1 - AllowKey=system.run[*] (AllowKeyRegexp=^system\.

Wartość domyślna: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

HeartbeatFrequency

Ustaw częstotliwość wiadomości heartbeat w sekundach. Używane do monitorowania dostępności aktywnych kontroli.
0 - wiadomości heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Określ interfejs hosta (adres IP lub nazwa DNS) używany podczas autorejestracji hosta. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawne skonfigurowanie adresu IP lub DNS. Więcej informacji znajdziesz w sekcji Using DNS as default interface.

Jeśli nie zostanie zdefiniowana, wartość zostanie pobrana z HostInterfaceItem.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Określ pozycję służącą do ustalenia interfejsu hosta (adresu IP lub nazwy DNS) podczas automatycznej rejestracji hosta. Ta wartość jest używana tylko wtedy, gdy parametr `HostInterface` nie jest zdefiniowany. Więcej informacji zawiera sekcja Używanie DNS jako domyślnego interfejsu.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja `system.run[]` jest obsługiwana niezależnie od ustawień `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

HostMetadata

Określ metadane używane do identyfikowania lub rozróżniania hosta podczas automatycznej rejestracji hosta (aktywny agent). `HostMetadata` umożliwia rozróżnianie hostów niezależnie od ich nazw.

Jeśli parametr nie zostanie zdefiniowany, wartość zostanie pobrana z `HostMetadataItem`.

agent zgłosi błąd i nie uruchomi się, jeśli określona wartość przekroczy limit 2034 bajtów lub będzie ciągiem znaków innym niż UTF-8.
Jeśli parametr oczekuje adresu IP lub nazwy DNS, wartości poprawne w UTF-8, ale niebędące prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Zakres: 0–2034 bajty

HostMetadataItem

Określ pozycję agenta Zabbix, aby uzyskać metadane hosta. Ta opcja jest używana tylko wtedy, gdy `HostMetadata` nie jest zdefiniowane.

Wartość `HostMetadataItem` jest pobierana przy każdej próbie automatycznej rejestracji i używana wyłącznie podczas procesu automatycznej rejestracji hosta (aktywny agent). `HostMetadataItem` umożliwia rozróżnianie hostów na podstawie innych informacji niż nazwa hosta.

Obsługiwane są parametry użytkownika, liczniki wydajności i aliasy. Pozycja `system.run[]` jest obsługiwana niezależnie od ustawień `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem znaków UTF-8; w przeciwnym razie zostanie zignorowana. Jeśli parametr oczekuje adresu IP lub nazwy DNS, wartości, które są prawidłowym tekstem UTF-8, ale nie są prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Nazwa hosta

Określ unikalne nazwy hostów rozróżniające wielkość liter. Lista rozdzielona przecinkami.
 Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze. Wartość jest pobierana z `HostnameItem`, jeśli nie została określona.
Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Określ pozycję agenta Zabbix służącą do pobierania nazwy hosta.
 Ta opcja jest używana tylko wtedy, gdy `Hostname` nie jest zdefiniowane.
 Parametry użytkownika, liczniki wydajności ani aliasy nie są obsługiwane, ale pozycja `system.run[]` jest obsługiwana niezależnie od wartości `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.
Więcej informacji zawiera szczegółowy opis.

Wartość domyślna: `system.hostname`

Include

Określ katalog lub pojedyncze pliki, które mają zostać uwzględnione w pliku konfiguracyjnym (domyślnie znajdującym się w `C:\Program Files\Zabbix Agent`, jeśli agent Zabbix został zainstalowany przy użyciu pakietów instalacyjnych Windows MSI; znajdującym się w folderze określonym podczas instalacji, jeśli agent Zabbix został zainstalowany jako archiwum zip).

Ogólne zasady dotyczące parametru `Include`:

- Ścieżka może być względna względem lokalizacji pliku `zabbix_agentd.conf`.
- W jednym pliku konfiguracyjnym można określić wiele parametrów `Include`.
- Zabbix przetwarza parametr `Include` natychmiast po jego odczytaniu, dlatego należy rozważyć umieszczenie go na końcu pliku konfiguracyjnego.

Zasady dotyczące uwzględnianych plików:

- Każdy plik musi być możliwy do odczytu.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie zostanie uruchomiony.

Zasady dotyczące uwzględniania katalogów lub wzorców plików:

- Nie należy zakładać żadnej kolejności uwzględniania (np. pliki nie są uwzględniane w kolejności alfabetycznej).
- Nie należy definiować tego samego parametru w więcej niż jednym uwzględnianym pliku (np. w celu zastąpienia ustawienia ogólnego ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Należy uważać na kopie zapasowe tworzone automatycznie przez niektóre edytory tekstu (np. edycja pliku `include/my_specific.co` może utworzyć plik `include/my_specific.conf.BAK`), ponieważ oba pliki zostaną uwzględnione.

Aby uwzględnić tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (`*`) do dopasowywania wzorców.

Przykład:

```
Include=C:\Program Files\Zabbix Agent\zabbix_agentd.d\*.conf
```

ListenBacklog

Ustaw maksymalną liczbę oczekujących połączeń w kolejce TCP.
 Wartość domyślna jest zakodowaną na stałe stałą, która zależy od systemu.
 Maksymalna obsługiwana wartość również zależy od systemu; zbyt wysokie wartości mogą zostać bez ostrzeżenia obcięte do „maksimum określonego przez implementację”.

Domyślnie: `SOMAXCONN`
 Zakres: 0 - INT_MAX

ListenIP

Określa adres IP, na którym agent nasłuchuje połączeń z serwera/proxy. Lista rozdzielana przecinkami.

Domyślnie: `0.0.0.0`

ListenPort

Określ port nasłuchiwania agenta dla połączeń z serwera/proxy.

Domyślnie: 10050
 Zakres: 1024-32767

LogFile

Określ plik dziennika agenta.

Domyślnie: `c:\zabbix_agentd.log`
 Wymagane: Tak, jeśli `LogType` jest ustawione na `file`; w przeciwnym razie nie

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
0 - wyłącza automatyczną rotację dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku z jakiegokolwiek powodu się nie powiedzie, istniejący plik dziennika zostanie obcięty i utworzony od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
 Polecenia będą rejestrowane tylko wtedy, gdy zostaną wykonane zdalnie.
 Wpisy w dzienniku nie będą tworzone, jeśli `system.run[]` zostanie uruchomione lokalnie przez parametry `HostMetadataItem`, `HostInterfaceItem` lub `HostnameItem`.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

LogType

Określ typ wyjścia logów: - *file* - zapisuj log do pliku określonego parametrem `LogFile`; - *system* - zapisuj log do Dziennika zdarzeń systemu Windows; - *console* - zapisuj log do standardowego wyjścia.

Domyślnie: `file`

MaxLinesPerSecond

Ustaw maksymalną liczbę nowych linii na sekundę wysyłanych do serwera/proxy Zabbix podczas przetwarzania aktywnych kontroli `log`, `logrt` lub `eventlog`.
 Podana wartość zostanie zastąpiona przez parametr `maxlines` określony w kluczu pozycja `log`, `logrt` lub `eventlog`.
Uwaga: Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w `MaxLinesPerSecond`, aby wyszukać wymagany ciąg w pozycjach dziennika.

Domyślnie: 20
 Zakres: 1-1000

PerfCounter

Zdefiniuj nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego przedziału czasu <period> (w sekundach).
Składnia: <parameter_name>,"<perf_counter_path>",<period>

Na przykład, jeśli chcesz otrzymywać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr "interrupts" w następujący sposób:

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycja podczas tworzenia pozycji. Próbki do obliczania średniej wartości będą pobierane co sekundę.
Możesz uruchomić `typeperf -qx`, aby uzyskać listę wszystkich liczników wydajności dostępnych w systemie Windows.

PerfCounterEn

Zdefiniuj nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego przedziału czasu <period> (w sekundach). W porównaniu do PerfCounter, ścieżki perfcounter muszą być podane w języku angielskim. Obsługiwane tylko w **Windows Server 2008/Vista** i nowszych.
Składnia: <parameter_name>,"<perf_counter_path>",<period>

Na przykład, jeśli chcesz otrzymywać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr "interrupts" w następujący sposób:

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycji podczas tworzenia pozycji. Próbki do obliczania średniej wartości będą pobierane co sekundę.
Listę angielskich ciągów można znaleźć, przeglądając następujący klucz rejestru: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

RefreshActiveChecks

Określa częstotliwość odświeżania listy aktywnych kontroli, w sekundach.
 Należy pamiętać, że po nieudanym odświeżeniu aktywnych kontroli następna próba odświeżenia zostanie podjęta po 60 sekundach.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Określ adresy IP, opcjonalnie w notacji CIDR, lub nazwy DNS serwerów/proxy Zabbix. Lista rozdzielana przecinkami.
 Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4. Należy pamiętać, że "IPv4-compatible IPv6 addresses" (`0000::/96` prefix) są obsługiwane, ale przestarzałe zgodnie z RFC4291. Dozwolone są spacje.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak, jeśli StartAgents nie jest jawnie ustawione na 0

ServerActive

Określ adres serwera/proxy Zabbix lub konfigurację klastra, aby pobierać aktywne kontrole z.
 Adres serwera/proxy jest adresem IP lub nazwą DNS oraz opcjonalnym portem oddzielonym dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwerów lub proxy oddzielonych średnikiem. Można podać wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy podawać więcej niż jednego proxy Zabbix z każdego serwera/klastra Zabbix. Jeśli podano proxy Zabbix, nie należy podawać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby używać kilku niezależnych serwerów Zabbix równolegle. Spacje są dozwolone.
Jeśli port nie zostanie określony, używany jest port domyślny.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla tego hosta określono port. Jeśli port nie jest określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne kontrole są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

Zakres: Liczba aktywnych serwerów wymienionych w `ServerActive` plus liczba wstępnie utworzonych instancji dla pasywnych kontroli określonych w `StartAgents` musi być mniejsza niż 64.

SourceIP

Określ adres IP źródłowy dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix;
- nawiązywania połączeń podczas wykonywania niektórych pozycji (`web.page.get`, `net.tcp.port` itp.).

StartAgents

Ustaw liczbę wstępnie utworzonych instancji `zabbix_agentd` do przetwarzania pasywnych kontroli.
 Jeśli ustawiono 0, pasywne kontrole są wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Domyślnie: 10
 Zakres: 0-100 (*)

Limit czasu

Określa, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr określa czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwanie na odpowiedź z serwera Zabbix;
- wysyłanie żądań do serwera Zabbix, w tym żądań konfiguracji pozycji oraz danych pozycji w active checks;
- pobieranie danych dziennika za pośrednictwem monitorowania pliku dziennika lub dziennika zdarzeń systemu Windows;
- wysyłanie komunikatów heartbeat;
- maksymalny czas trwania dla sprawdzeń `vfs.*`;
- używany przez moduły agent Zabbix;
- pełnienie roli wartości awaryjnej w scenariuszach, w których serwer lub proxy w wersji starszej niż 7.0 wysyła sprawdzenia bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych sprawdzeń agent, które mają konfigurowalne ustawienia limitu czasu w frontendzie (na poziomie globalnym, proxy lub dla pojedynczej pozycji).

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Określ poziom szyfrowania wymagany dla połączeń przychodzących.
 Używane dla kontroli pasywnych.
 Można podać wiele wartości, oddzielając je przecinkiem: - *unencrypted* - akceptuj połączenia bez szyfrowania (domyślnie); - *psk* - akceptuj połączenia z TLS i współdzielonym kluczem wstępnym (PSK); - *cert* - akceptuj połączenia z TLS i certyfikatem.

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używaną do szyfrowanej komunikacji między komponentami Zabbixa.

TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używany do szyfrowanej komunikacji z komponentami Zabbix.

TLSConnect

Określ poziom szyfrowania wymagany dla połączeń agent z serwerem Zabbix lub proxy.
 Używane dla aktywnych kontroli.
 Można określić tylko jedną wartość: - *unencrypted* - połączenie bez szyfrowania (domyślnie); - *psk* - połączenie z użyciem TLS i wstępnie współdzielonego klucza (PSK); - *cert* - połączenie z użyciem TLS i certyfikatu.

Obowiązkowe: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz agenta, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Określ pełną ścieżkę do pliku zawierającego współdzielony klucz agenta, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSPSKIdentity

Określ ciąg tożsamości pre-shared key, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSServerCertIssuer

Określ dozwolonego wystawcę certyfikatu serwera (proxy).

TLSServerCertSubject

Określa dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Dodatkowo znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

UserParameter

Określ zdefiniowany przez użytkownika parametr do monitorowania. Może istnieć kilka takich parametrów.
Format: `UserParameter=<key>,<shell command>`
Należy pamiętać, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie znaku końca linii. Polecenia powłoki mogą używać ścieżek względnych, jeśli określono parametr `UserParameterDir`.

Przykład:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Określa domyślną ścieżkę wyszukiwania dla poleceń `UserParameter`. Jeśli jest używana, agent zmieni swój katalog roboczy na wskazany tutaj przed wykonaniem polecenia. Dzięki temu polecenia `UserParameter` mogą mieć względny prefiks `./` zamiast pełnej ścieżki. Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=C:\opt\myscripts
```

Zabbix agent w Microsoft Windows

Konfigurowanie agent

Obie generacje agentów Zabbix działają jako usługa systemu Windows. W przypadku Zabbix agent 2 zamień `agentd` na `agent2` w poniższych instrukcjach.

Na hoście Microsoft Windows można uruchomić pojedynczą instancję agenta Zabbix lub wiele instancji agenta. Pojedyncza instancja może używać:

- domyślnego pliku konfiguracyjnego znajdującego się w tym samym katalogu co plik binarny agenta;
- pliku konfiguracyjnego określonego w wierszu poleceń.

W przypadku wielu instancji każda instancja agenta musi mieć własny plik konfiguracyjny (jedna z instancji może używać domyślnego pliku konfiguracyjnego).

Przykładowy plik konfiguracyjny jest dostępny w archiwum źródłowym Zabbix jako:

- `conf/zabbix_agentd.conf` dla Zabbix agent;
- `conf/zabbix_agent2.conf` dla Zabbix agent2.

Jeśli chcesz zainstalować Zabbix agent/agent 2 dla systemu Windows jako usługę z archiwum bez jawnego określania pliku konfiguracyjnego, przed zainstalowaniem agenta:

- `conf/zabbix_agentd.conf` należy ręcznie skopiować do katalogu, w którym zostanie zainstalowany plik `zabbix_agentd.exe`;
- `conf/zabbix_agent2.conf` oraz katalog `conf/zabbix_agent2.d` należy ręcznie skopiować do katalogu, w którym zostanie zainstalowany plik `zabbix_agent2.exe`.

Szczegółowe informacje dotyczące konfigurowania agenta Zabbix dla systemu Windows zawierają opcje pliku konfiguracyjnego agenta Zabbix lub pliku konfiguracyjnego agenta 2 Zabbix.

Parametr Hostname

Aby wykonywać aktywne sprawdzanie na hoście, Zabbix agent musi mieć zdefiniowaną nazwę hosta. Ponadto wartość nazwy hosta ustawiona po stronie agenta powinna dokładnie odpowiadać wartości *Nazwa hosta* skonfigurowanej dla hosta w frontend.

Wartość nazwy hosta po stronie agenta można zdefiniować za pomocą parametru `Hostname` lub `HostnameItem` w pliku konfiguracyjnym agenta - w przeciwnym razie używane są wartości domyślne, jeśli żaden z tych parametrów nie został określony.

Wartością domyślną parametru `HostnameItem` jest wartość zwracana przez klucz agenta `system.hostname`. W systemie Windows zwraca on wynik funkcji `gethostname()`, która odpytuje dostawców przestrzeni nazw w celu określenia lokalnej nazwy hosta. Jeśli żaden dostawca przestrzeni nazw nie odpowie, zwracana jest nazwa NetBIOS.

Wartością domyślną parametru `Hostname` jest wartość zwracana przez parametr `HostnameItem`. W praktyce oznacza to, że jeśli oba te parametry nie zostaną określone, rzeczywistą nazwą hosta będzie nazwa NetBIOS hosta; Zabbix agent użyje nazwy hosta NetBIOS do pobrania listy aktywnych sprawdzeń z serwera Zabbix i prześle do niego wyniki.

Klucz `system.hostname` obsługuje dwa opcjonalne parametry - *type* i *transform*.

Type określa typ nazwy, którą powinna zwrócić pozycja:

- *netbios* (domyślnie) - zwraca nazwę hosta NetBIOS, która jest ograniczona do 15 znaków i zawiera wyłącznie wielkie litery;
- *host* - rozróżnia wielkość liter, zwraca pełną, rzeczywistą nazwę hosta Windows (bez domeny);
- *shorthost* - zwraca część nazwy hosta znajdującą się przed pierwszą kropką. Jeśli nazwa nie zawiera kropki, zwracany jest cały ciąg znaków.
- *fqdn* - zwraca w pełni kwalifikowaną nazwę domenową (Fully Qualified Domain Name) bez końcowej kropki.

Transform umożliwia określenie dodatkowej reguły przekształcania nazwy hosta:

- *none* (domyślnie) - używa oryginalnej wielkości liter;
- *lower* - konwertuje tekst na małe litery.

Aby uprościć konfigurację pliku `zabbix_agentd.conf` i ujednolicić ją, można zastosować różne podejścia:

1. Pozostawić parametry `Hostname` lub `HostnameItem` niezdefiniowane - Zabbix agent użyje nazwy hosta NetBIOS jako nazwy hosta.
2. Pozostawić parametr `Hostname` niezdefiniowany i zdefiniować `HostnameItem` w następujący sposób:
   - `HostnameItem=system.hostname[host]` - aby Zabbix agent używał pełnej, rzeczywistej nazwy hosta Windows (z rozróżnianiem wielkości liter) jako nazwy hosta.
   - `HostnameItem=system.hostname[shorthost,lower]` - aby Zabbix agent używał tylko części nazwy hosta znajdującej się przed pierwszą kropką, przekonwertowanej na małe litery.
   - `HostnameItem=system.hostname[fqdn]` - aby Zabbix agent używał w pełni kwalifikowanej nazwy domenowej (Fully Qualified Domain Name) jako nazwy hosta.

Nazwa hosta jest również używana jako część nazwy usługi systemu Windows, która służy do instalowania, uruchamiania, zatrzymywania i odinstalowywania usługi systemu Windows. Na przykład jeśli plik konfiguracyjny Zabbix agent określa `Hostname=Windows_db_server`, agent zostanie zainstalowany jako usługa systemu Windows `Zabbix Agent [Windows_db_server]`. Dlatego aby każda instancja Zabbix agent miała inną nazwę usługi systemu Windows, każda instancja musi używać innej nazwy hosta.

Instalowanie agenta jako usługi systemu Windows

Przed zainstalowaniem agenta skopiuj ręcznie `conf/zabbix_agentd.conf` do katalogu, w którym zostanie zainstalowany `zabbix_agentd.exe`.

Aby zainstalować pojedynczą instancję agenta Zabbix z domyślnym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --install
```

Attention:
W systemie 64-bitowym do poprawnego działania wszystkich kontroli związanych z uruchamianiem procesów 64-bitowych wymagana jest 64-bitowa wersja agenta Zabbix.

Jeśli chcesz użyć innego pliku konfiguracyjnego niż domyślny, powinieneś użyć następującego polecenia do instalacji usługi:

```
zabbix_agentd.exe --config <your_configuration_file> --install
```

Należy podać pełną ścieżkę do pliku konfiguracyjnego.

Wiele instancji agenta Zabbix można zainstalować jako usługi w następujący sposób:

```
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
```

Zainstalowana usługa powinna być teraz widoczna w Panelu sterowania.

Uruchamianie agent

Aby uruchomić usługę agenta, możesz użyć Panelu sterowania lub zrobić to z wiersza poleceń.

Aby uruchomić pojedynczą instancję Zabbix agent z domyślnym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --start
```

Aby uruchomić pojedynczą instancję Zabbix agent z innym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --config <your_configuration_file> --start
```

Aby uruchomić jedną z wielu instancji Zabbix agent:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --start --multiple-agents
```

Zatrzymywanie agent

Aby zatrzymać usługę agenta, możesz użyć Panelu sterowania lub wykonać to z wiersza poleceń.

Aby zatrzymać pojedynczą instancję agenta Zabbix uruchomioną z domyślnym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --stop
```

Aby zatrzymać pojedynczą instancję agenta Zabbix uruchomioną z innym plikiem konfiguracyjnym:

```
zabbix_agentd.exe --config <your_configuration_file> --stop
```

Aby zatrzymać jedną z wielu instancji agenta Zabbix:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --stop --multiple-agents
```

Odinstalowywanie usługi Windows agent

Aby odinstalować pojedynczą instancję Zabbix agent przy użyciu domyślnego pliku konfiguracyjnego:

```
zabbix_agentd.exe --uninstall
```

Aby odinstalować pojedynczą instancję Zabbix agent przy użyciu niestandardowego pliku konfiguracyjnego:

```
zabbix_agentd.exe --config <your_configuration_file> --uninstall
```

Aby odinstalować wiele instancji Zabbix agent z usług 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
```

Ograniczenia

Agent Zabbix dla Windows nie obsługuje niestandardowych konfiguracji Windows, w których procesory są rozmieszczone nierównomiernie między węzłami NUMA. Jeśli logiczne procesory są rozmieszczone nierównomiernie, metryki wydajności procesora mogą nie być dostępne dla niektórych procesorów. Na przykład, jeśli istnieje 72 logicznych procesorów i 2 węzły NUMA, oba węzły muszą mieć po 36 procesorów każdy.

agent Zabbix w systemie IBM AIX

Przegląd

Wstępnie skompilowane pliki binarne agent dla IBM AIX są dostępne do pobrania jako archiwum `.tar.gz`. Na tej stronie opisano instalację agent z archiwum, uruchamianie go ręcznie lub za pomocą System Resource Controller (SRC), uruchamianie go podczas rozruchu systemu oraz usuwanie go z SRC.

Instalowanie agent

Polecenie `tar` w systemie AIX nie obsługuje opcji `-z`, dlatego archiwum należy rozpakować za pomocą `gunzip`. Użyj opcji `-o` polecenia `tar`, aby wyodrębnione pliki były własnością użytkownika, który je wyodrębnia, a nie użytkowników i grup o identyfikatorach numerycznych zapisanych w archiwum. Na przykład, aby umieścić agent w katalogu `/opt/zabbix` (uruchom jako root; katalog `/opt/zabbix` nie może jeszcze istnieć):

```
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
```

W archiwum AIX plik binarny agent znajduje się w katalogu `usr/sbin/`, a plik konfiguracyjny w katalogu `etc/zabbix/`. Przed uruchomieniem agent dostosuj plik `/opt/zabbix/etc/zabbix/zabbix_agentd.conf` zgodnie z potrzebami (co najmniej parametry `Server`, `ServerActive` i `Hostname`).

Jeśli agent zostanie uruchomiony jako root, przełączy się na użytkownika `zabbix` (lub użytkownika określonego za pomocą parametru `User`), chyba że w pliku konfiguracyjnym ustawiono `AllowRoot=1`. Ten użytkownik musi istnieć w systemie; w przeciwnym razie agent nie zostanie uruchomiony. Jeśli użytkownik `zabbix` jeszcze nie istnieje, utwórz go wraz z grupą o tej samej nazwie:

```
mkgroup zabbix
mkuser pgrp=zabbix login=false rlogin=false zabbix
```

Atrybuty `login=false` i `rlogin=false` uniemożliwiają używanie konta do logowania; agent nadal może działać jako ten użytkownik.

Ręczne uruchamianie agenta

Agenta można uruchomić ręcznie:

```
/opt/zabbix/usr/sbin/zabbix_agentd -c /opt/zabbix/etc/zabbix/zabbix_agentd.conf
```

Uruchamianie agenta z SRC

Rejestrowanie agenta

Agenta można również zarejestrować w kontrolerze zasobów systemowych (SRC). Po uruchomieniu za pośrednictwem SRC agent musi działać w trybie pierwszoplanowym (opcja `-f`). W przeciwnym razie SRC traci możliwość śledzenia procesu agenta i nie może go zatrzymać. Na przykład (uruchom jako 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
```

W tym przykładzie:

- `-s` ustawia nazwę podsystemu używaną wraz z poleceniami SRC;
- `-p` ustawia pełną ścieżkę do pliku binarnego agenta;
- `-a` ustawia argumenty wiersza poleceń agenta jako pojedynczy ciąg ujęty w cudzysłowy; SRC dzieli ten ciąg na osobne argumenty podczas uruchamiania agenta (akceptowana jest tylko jedna opcja `-a`);
- `-u` ustawia użytkownika, z którego uprawnieniami działa agent; nazwa użytkownika jest konwertowana na numeryczny identyfikator użytkownika podczas tworzenia podsystemu, dlatego użytkownik musi już istnieć;
- `-G` dodaje agenta do grupy podsystemów `tcpip`, dzięki czemu dotyczą go również polecenia grupowe (na przykład `stopsrc -g tcpip` zatrzymuje także agenta);
- `-S -n 15 -f 9` powoduje, że SRC steruje agentem za pomocą sygnałów: `stopsrc` wysyła `SIGTERM` (15) w celu normalnego zatrzymania, a `SIGKILL` (9) w celu wymuszenia zatrzymania (`stopsrc -f`);
- `-R` powoduje, że SRC ponownie uruchamia agenta, jeśli zakończy on działanie w nieprawidłowy sposób (na przykład w wyniku awarii jednego z jego procesów); agent nie jest uruchamiany ponownie po wykonaniu `stopsrc`.

Ponieważ identyfikator użytkownika jest przechowywany w definicji podsystemu, zaktualizuj tę definicję, jeśli użytkownik `zabbix` zostanie utworzony ponownie z innym identyfikatorem użytkownika:

```
chssys -s zabbix_agentd -u zabbix
```

Uruchamianie i zatrzymywanie agent

Agent można uruchomić, sprawdzić jego stan i zatrzymać za pomocą poleceń `startsrc`, `lssrc` i `stopsrc`:

```
startsrc -s zabbix_agentd
lssrc -s zabbix_agentd
stopsrc -s zabbix_agentd
```

Unikaj wymuszonego zatrzymania (`stopsrc -f`). Kończy ono tylko główny proces agent; pozostałe procesy agent nadal działają i utrzymują otwarty port nasłuchiwania, dlatego agent nie może zostać ponownie uruchomiony, dopóki procesy te nie zostaną ręcznie zakończone (na przykład za pomocą `kill -9`).

Uruchamianie agent przy starcie systemu

Zarejestrowanie agent w SRC nie powoduje jego uruchamiania przy starcie systemu. Aby automatycznie uruchamiać agent podczas rozruchu, dodaj wpis do pliku `/etc/inittab`:

```
mkitab "zabbix_agentd:2:once:/usr/bin/startsrc -s zabbix_agentd >/dev/console 2>&1"
```

Usuwanie agent z SRC

Aby usunąć agent z uruchamiania systemowego oraz z SRC:

```
stopsrc -s zabbix_agentd
rmitab zabbix_agentd
rmssys -s zabbix_agentd
```

Agent 2

Omówienie Zabbix agent 2 to nowa generacja Zabbix agent, napisana w Go (z wykorzystaniem części kodu C odziedziczonego po Zabbix agent). Została zaprojektowana, aby:

- Zmniejszyć liczbę połączeń TCP;
- Zapewnić lepszą współbieżność kontroli;
- Być łatwo rozszerzalna za pomocą pluginów, które zapewniają proste kontrole przy minimalnej ilości kodu oraz obsługują złożone kontrole składające się z długo działających skryptów i samodzielnego zbierania danych z okresowym raportowaniem;
- Działać jako zamiennik Zabbix agent, obsługując wszystkie wcześniejsze funkcje.

Wszystkie metryki w Zabbix agent 2 są zbierane przez pluginy.

Te funkcje są specyficzne dla pluginów:

- Zaplanowane i elastyczne interwały zarówno dla kontroli pasywnych, jak i aktywnych
- Zarządzanie kolejką zadań z uwzględnieniem harmonogramu i współbieżności zadań
- Limity czasu na poziomie pluginu
- Sprawdzanie zgodności Zabbix agent 2 i jego pluginów podczas uruchamiania

Istnieją dwa typy pluginów:

- Wbudowane pluginy
- Ładowalne pluginy

Możesz także tworzyć własne pluginy.

Własnoręcznie utworzone pluginy nie muszą być wbudowane w agent 2. Możesz dodać je jako ładowalne pluginy, co upraszcza tworzenie pluginów dla nowych metryk.

Ładowalne pluginy można dodawać w czasie działania bez ponownej kompilacji Zabbix agent 2.

Zobacz także:

- Porównanie agent i agent 2
- Parametry konfiguracji: Zabbix agent 2
- Parametry konfiguracji: Zabbix agent 2 (Windows)

- Parametry konfiguracji: Pluginy
- Protokół pluginu Zabbix agent 2

Sprawdzanie pasywne i aktywne

Zabbix agent 2 obsługuje sprawdzanie pasywne i aktywne, podobnie jak Zabbix agent. Dodatkowo aktywne sprawdzanie przez Zabbix agent 2 obsługuje elastyczne interwały i interwały harmonogramu oraz współbieżność sprawdzania w ramach jednego aktywnego serwera.

> **Note:**
> Domyślnie po ponownym uruchomieniu Zabbix agent 2 zaplanuje pierwsze zbieranie danych dla aktywnego sprawdzania w warunkowo losowym momencie w ramach interwału aktualizacji pozycji, aby zapobiec skokom wykorzystania zasobów. Aby wykonywać aktywne sprawdzanie, które nie ma interwału aktualizacji *Scheduling* interwału aktualizacji, natychmiast po ponownym uruchomieniu agenta, ustaw parametr `ForceActiveChecksOnStart` (na poziomie globalnym) lub parametr wtyczki `Plugins.<Plugin name>.System.ForceActiveChecksOnStart` parametr wtyczki (dotyczy tylko sprawdzania określonych wtyczek) w pliku konfiguracyjnym. Jeśli parametr na poziomie wtyczki jest ustawiony, zastąpi parametr globalny.

Sprawdzanie współbieżności

Sprawdzenia z różnych wtyczek mogą być wykonywane równolegle. Liczba równoczesnych sprawdzeń w ramach jednej wtyczki jest ograniczona przez ustawienie pojemności wtyczki. Każda wtyczka może mieć zakodowane na stałe ustawienie pojemności (1000 jako wartość domyślna), które można obniżyć za pomocą ustawienia `Plugins.<PluginName>.System.Capacity=N` w parametrze konfiguracji `Plugins`.

Obsługiwane platformy Informacje o obsługiwanych platformach znajdują się na stronie Wymagania.

Agent 2 w systemach uniksopodobnych Zabbix agent 2 w systemach uniksopodobnych jest uruchamiany na host monitorowanym.

Instalacja

Zabbix agent 2 można zainstalować w systemach opartych na Linux przy użyciu jednej z następujących metod:

- Pakiety Zabbix - wybierz komponent Agent 2 (po wybraniu wersji Zabbix, dystrybucji systemu operacyjnego i wersji systemu operacyjnego) i postępuj zgodnie z instrukcjami;
- Źródła Zabbix - pobierz pliki źródłowe i skompiluj agent, konfigurując opcję `--enable-agent2`.

> **Note:**
> Możliwości monitorowania Zabbix agent 2 można rozszerzyć za pomocą ładowalnych wtyczek, które są dostępne osobno. Szczegóły można znaleźć w sekcji Ładowalne wtyczki.

Jeśli zainstalowano jako pakiet

Zabbix agent 2 działa jako proces pierwszoplanowy i do działania w tle korzysta z zewnętrznego menedżera usług (np. systemd). Zabbix agent 2 nie ma wbudowanej obsługi demonizacji w systemie Linux.

Agent można uruchomić, wykonując:

```
systemctl start zabbix-agent2
```

Aby zatrzymać, uruchomić ponownie lub sprawdzić stan Zabbix agent 2, użyj następujących poleceń:

```
systemctl stop zabbix-agent2
systemctl restart zabbix-agent2
systemctl status zabbix-agent2
```

Uruchomienie ręczne

Możesz uruchomić agent Zabbix, lokalizując plik binarny `zabbix_agent2` i uruchamiając go bezpośrednio; na przykład:

```
zabbix_agent2
```

Agent 2 w systemach Windows Zabbix agent 2 działa jako samodzielny proces; może jednak być również uruchamiany jako usługa systemu Windows.

Instalacja

Zabbix agent 2 można zainstalować w systemie Windows jedną z następujących metod:

- Wstępnie skompilowane binaria Zabbix agent - pobierz pakiet instalacyjny MSI agent i postępuj zgodnie z instrukcjami na stronie Instalacja agenta Windows z pliku MSI;
- Źródła Zabbix - pobierz pliki źródłowe i postępuj zgodnie z instrukcjami na stronie Kompilacja Zabbix agent 2 w systemie Windows.

Note:
Możliwości monitorowania Zabbix agent 2 można rozszerzyć za pomocą ładowalnych wtyczek, które są dostępne osobno. Szczegóły znajdziesz na stronie Ładowalne wtyczki.

Dodatkowe informacje dotyczące instalacji Zabbix agent 2 (z archiwum ZIP) jako usługi systemu Windows znajdziesz na stronie Zabbix agent w Microsoft Windows.

Opcje Z agentem Zabbix 2 można używać następujących parametrów wiersza poleceń:

Parametr	Opis
agent UNIX i Windows	
`-c --config <config-file>`	Ścieżka do pliku konfiguracyjnego. Tej opcji można użyć do określenia pliku konfiguracyjnego innego niż domyślny.
`-f --foreground`	Uruchomienie agenta Zabbix na pierwszym planie (domyślnie: true).
`-p --print`	Wyświetlenie znanych pozycji i zakończenie działania. Aby zwrócić również wyniki parametru użytkownika, należy określić plik konfiguracyjny (jeśli nie znajduje się on w domyślnej lokalizacji).
`-t --test <item key>`	Przetestowanie określonej pozycji i zakończenie działania. Aby zwrócić również wyniki parametru użytkownika, należy określić plik konfiguracyjny (jeśli nie znajduje się on w domyślnej lokalizacji).
`-T --test-config`	Sprawdzenie poprawności pliku konfiguracyjnego i zakończenie działania.
`-h --help`	Wyświetlenie informacji pomocy i zakończenie działania.
`-v --verbose`	Wyświetlenie informacji debugowania. Tej opcji należy używać razem z opcjami `-p` i `-t`.
`-V --version`	Wyświetlenie wersji agenta i informacji o licencji.
`-R --runtime-control <option>`	Wykonanie funkcji administracyjnych. Zobacz sterowanie w czasie działania.
Tylko agent Windows	
`-m --multiple-agents`	Używanie wielu instancji agenta (z opcjami `-i`, `-d`, `-s`, `-x`). Aby rozróżnić nazwy usług poszczególnych instancji, każda nazwa usługi będzie zawierać wartość Hostname z określonego pliku konfiguracyjnego.
`-S --startup-type <value>`	Ustawienie typu uruchamiania usługi agenta Zabbix dla systemu Windows. Dozwolone wartości: `automatic` - *(domyślnie)* automatyczne uruchamianie usługi podczas uruchamiania systemu Windows; `delayed` - opóźnienie uruchomienia usługi do czasu zakończenia uruchamiania usług uruchamianych automatycznie; `manual` - ręczne uruchamianie usługi (przez użytkownika lub aplikację); `disabled` - wyłączenie usługi, aby nie mogła zostać uruchomiona przez użytkownika ani aplikację. Tej opcji można używać razem z opcją `-i` lub osobno, aby zmodyfikować typ uruchamiania już zainstalowanej usługi.
`-i --install`	Zainstalowanie agenta Zabbix dla systemu Windows jako usługi.
`-d --uninstall`	Odinstalowanie usługi agenta Zabbix dla systemu Windows.
`-s --start`	Uruchomienie usługi agenta Zabbix dla systemu Windows.
`-x --stop`	Zatrzymanie usługi agenta Zabbix dla systemu Windows.

Poniżej przedstawiono przykłady użycia parametrów wiersza poleceń:

- Wyświetlenie wszystkich wbudowanych pozycji agenta wraz z wartościami.
- Przetestowanie parametru użytkownika z kluczem `mysql.ping` zdefiniowanym w określonym pliku konfiguracyjnym.
- Zainstalowanie usługi agenta Zabbix dla systemu Windows przy użyciu domyślnej ścieżki do pliku konfiguracyjnego.
- Zmodyfikowanie typu uruchamiania zainstalowanej usługi agenta Zabbix dla systemu Windows przy użyciu pliku konfiguracyjnego znajdującego się w tym samym folderze co plik wykonywalny agenta.

```
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
```

Sterowanie w czasie działania

Sterowanie w czasie działania udostępnia opcje zdalnego sterowania.

Option	Description
log_level_increase	Zwiększenie poziomu rejestrowania.
log_level_decrease	Zmniejszenie poziomu rejestrowania.
metrics	Wyświetlenie listy dostępnych metryk.
periodic_prof_enable	Włączenie profilowania okresowego. Wszystkie metryki zbierają dane w danym momencie i zapisują je, z wyjątkiem danych procesora. Okresowe dane procesora są zbierane w czasie do ukrytego pliku tymczasowego. Po zakończeniu interwału plik jest publikowany jako profil procesora, a skonfigurowana liczba ukończonych plików jest zachowywana. Plik tymczasowy nie jest uwzględniany w wartości ProfilerMaxFilesPerProfile.
periodic_prof_disable	Wyłączenie profilowania okresowego.
periodic_prof_execute	Zapisanie profili po 5-sekundowym próbkowaniu. Gdy profilowanie okresowe jest włączone, to polecenie kończy bieżący interwał procesora, zapisuje osobny 5-sekundowy profil procesora i rozpoczyna nowy okresowy plik procesora. Profile jawnie żądane są rotowane zgodnie z wartością ProfilerMaxFilesPerProfile po zakończeniu polecenia. Podczas zbierania profili przez to polecenie wszystkie polecenia profilera w czasie działania są odrzucane z komunikatem „profiler is busy”.
periodic_prof_set_interval	Ustawienie interwału profilowania.
version	Wyświetlenie wersji agent.
userparameter_reload	Ponowne wczytanie wartości opcji UserParameter i Include z bieżącego pliku konfiguracyjnego.
help	Wyświetlenie informacji pomocy dotyczących sterowania w czasie działania.

Przykłady:

- Zwiększenie poziomu rejestrowania dla agent 2.
- Wyświetlenie opcji sterowania w czasie działania.

```
zabbix_agent2 -R log_level_increase
zabbix_agent2 -R help
```

Plik konfiguracyjny Domyślna ścieżka do pliku konfiguracyjnego zależy od systemu operacyjnego i metody instalacji Zabbix:

- W systemie UNIX domyślna ścieżka to:
  - W przypadku instalacji ze źródeł: /usr/local/etc/zabbix_agent2.conf lub ścieżka określona przez zmienne czasu kompilacji --sysconfdir lub --prefix.
  - W przypadku instalacji z pakietów: /etc/zabbix/zabbix_agent2.conf.
- W systemie Windows domyślna ścieżka to C:\Program Files\Zabbix Agent 2\zabbix_agent2.conf.

Jeśli na hoście działa pojedyncza instancja agenta, może ona używać domyślnego pliku konfiguracyjnego lub pliku konfiguracyjnego określonego w wierszu poleceń. W przypadku wielu instancji każda instancja agenta musi mieć własny plik konfiguracyjny (jedna z instancji może używać domyślnego pliku konfiguracyjnego).

Parametry konfiguracyjne agenta Zabbix 2 są w większości zgodne z parametrami agenta Zabbix, z pewnymi wyjątkami:

Parametry		Opis
Parametry konfiguracyjne specyficzne dla agenta Zabbix 2		
	ControlSocket	Ścieżka do gniazda sterowania w czasie działania. Agent Zabbix 2 używa gniazda sterowania do wykonywania poleceń w czasie działania.
	EnablePersistentBuffer, PersistentBufferFile, PersistentBufferPeriod	Parametry te służą do konfigurowania trwałego przechowywania danych w agencie Zabbix 2 dla aktywnych pozycji.
	ForceActiveChecksOnStart	Określa, czy agent powinien wykonywać aktywne sprawdzenia natychmiast po ponownym uruchomieniu, czy rozłożyć je równomiernie w czasie.
	Plugins	Wtyczki mogą mieć własne parametry w formacie Plugins.<Plugin name>.<Parameter>=<value>. Typowym parametrem wtyczki jest System.Capacity, który określa limit sprawdzeń wykonywanych jednocześnie.
	StatusPort	Port, na którym agent Zabbix 2 będzie nasłuchiwać żądań stanu HTTP oraz wyświetlać listę skonfigurowanych wtyczek i niektórych parametrów wewnętrznych.
Parametry konfiguracyjne agenta Zabbix, które nie są obsługiwane przez agenta Zabbix 2		
	AllowRoot, User	Nie są obsługiwane przez agenta Zabbix 2, ponieważ nie obsługuje on działania jako demon.
	EnableRemoteCommands	Przestarzały; w obu agentach zamiast niego używane są parametry AllowKey, AllowKeyRegexp, DenyKey i DenyKeyRegexp.
	ListenBacklog	W agencie Zabbix służy do określania maksymalnej liczby oczekujących połączeń w kolejce TCP. Nie jest obsługiwany przez agenta Zabbix 2.
	LoadModule, LoadModulePath	Moduły ładowalne nie są obsługiwane przez agenta Zabbix 2.
	MaxLinesPerSecond, LogRemoteCommands	W agencie Zabbix 2 zamiast nich używane są Plugins.Log.MaxLinesPerSecond i Plugins.SystemRun.LogRemoteCommands.

Parametry	Opis
`StartAgents`	Parametr ten jest używany w agencie Zabbix do zwiększania współbieżności pasywnych sprawdzeń lub ich wyłączania. W agencie Zabbix 2 współbieżność jest konfigurowana na poziomie wtyczki i może być ograniczana ustawieniem pojemności. Pasywne sprawdzenia są wyłączone w agencie Zabbix 2, jeśli parametr `Server` nie został określony.

Więcej informacji można znaleźć w opcjach pliku konfiguracyjnego dla agenta Zabbix 2.

Kody wyjścia Zabbix agent 2 może być również skompilowany ze starszymi wersjami OpenSSL (1.0.1, 1.0.2).

W takim przypadku Zabbix udostępnia mutexy do blokowania w OpenSSL. Jeśli operacja zablokowania lub odblokowania mutexa zakończy się niepowodzeniem, komunikat o błędzie jest wypisywany do standardowego strumienia błędów (STDERR), a Agent 2 kończy działanie z kodem zwrotnym 2 lub 3, odpowiednio.

Porównanie Agent vs agent 2

Ta sekcja opisuje różnice między Zabbix agent i Zabbix agent 2.

Parameter	Zabbix agent	Zabbix agent 2
Język programowania	C	Go z niektórymi częściami w C
Demonizacja	tak	tylko przez systemd (tak w Windows)
Obsługiwane rozszerzenia	Niestandardowe moduły ładowalne w C.	Niestandardowe wtyczki w Go.
Wymagania		
Obsługiwane platformy	Linux, IBM AIX, FreeBSD, NetBSD, OpenBSD, HP-UX, Mac OS X, Solaris: 9, 10, 11, Windows: wszystkie wersje desktopowe i serwerowe od XP	Linux, Windows: wszystkie wersje desktopowe i serwerowe, na których można zainstalować obsługiwaną wersję Go.
Obsługiwane biblioteki kryptograficzne	GnuTLS 3.1.18 i nowsze. OpenSSL 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x. LibreSSL - testowano z wersjami 2.7.4, 2.8.2 (obowiązują pewne ograniczenia, zobacz stronę Encryption, aby uzyskać szczegóły).	Linux: OpenSSL 1.0.1 i nowszy. MS Windows: OpenSSL 1.1.1 lub nowszy. Biblioteka OpenSSL musi mieć włączoną obsługę PSK. LibreSSL nie jest obsługiwany.
Monitorowanie procesów		
Procesy	Oddzielny proces aktywnego sprawdzania dla każdego rekordu serwer/proxy.	Jeden proces z automatycznie tworzonymi wątkami. Maksymalna liczba wątków jest określana przez zmienną środowiskową `GOMAXPROCS`.
Metryki	**UNIX**: zobacz listę obsługiwanych pozycji. **Windows**: zobacz listę dodatkowych, specyficznych dla Windows pozycji.	**UNIX**: Wszystkie metryki obsługiwane przez Zabbix agent. Ponadto agent 2 zapewnia natywne dla Zabbix rozwiązanie monitorowania dla: Docker, Memcached, MySQL, PostgreSQL, Redis, systemd i innych celów monitorowania - zobacz pełną listę specyficznych dla agent 2 pozycji. **Windows**: Wszystkie metryki obsługiwane przez Zabbix agent, a także kontrole `net.tcp.service*` dla HTTPS, LDAP. Ponadto agent 2 zapewnia natywne dla Zabbix rozwiązanie monitorowania dla: PostgreSQL, Redis.
Współbieżność	Aktywne sprawdzenia dla pojedynczego serwer są wykonywane sekwencyjnie.	Sprawdzenia z różnych wtyczek lub wiele sprawdzeń w ramach jednej wtyczki mogą być wykonywane współbieżnie.
Trapy firm trzecich	nie	tak

Parameter	Zabbix agent	Zabbix agent 2
Dodatkowe funkcje		
Trwała pamięć masowa	nie	tak
Trwałe pliki dla metryk `log*[]`	tak (tylko w Unix)	nie
Przesyłanie danych logów	Może być wykonywane podczas zbierania logów, aby zwolnić bufor.	Zbieranie logów jest zatrzymywane, gdy bufor jest pełny, dlatego parametr `BufferSize` musi być co najmniej równy `MaxLinesPerSecond` x 2.
Zmienia użytkownika w czasie działania	tak (tylko systemy podobne do Unix)	nie (kontrolowane przez systemd)
Konfigurowalne przez użytkownika zestawy szyfrów	tak	nie

Zobacz także:

- *Opis procesów Zabbix*: Zabbix agent, Zabbix agent 2
- *Parametry konfiguracji*: Zabbix agent UNIX / Windows, Zabbix agent 2 UNIX / Windows

Wbudowane wtyczki

Te wtyczki są wbudowane w Zabbix agent 2 i są dostępne od razu po instalacji. Kliknij nazwę wtyczki, aby przejść do repozytorium wtyczki z dodatkowymi informacjami.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Uwagi
Agent	Metryki używanego Zabbix agent.	agent.hostname, agent.ping, agent.version	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
CPU	Monitorowanie procesora systemu (liczba procesorów/rdzeni CPU, wykryte procesory, procent wykorzystania).	system.cpu.discovery, system.cpu.num, system.cpu.util	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Docker	Monitorowanie kontenerów Docker.	docker.container_info, docker.container_stats, docker.containers, docker.containers.discovery, docker.data_usage, docker.images, docker.images.discovery, docker.info, docker.ping	Zobacz także: Parametry konfiguracji
File	Zbieranie metryk plików.	vfs.file.cksum, vfs.file.contents, vfs.file.exists, vfs.file.md5sum, vfs.file.regexp, vfs.file.regmatch, vfs.file.size, vfs.file.time	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Kernel	Monitorowanie jądra systemu.	kernel.maxfiles, kernel.maxproc	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Uwagi
Log	Monitorowanie plików dziennika.	log, log.count, logrt, logrt.count	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys. Zobacz także: Parametry konfiguracji wtyczki (Unix/Windows)
Memcached	Monitorowanie serwera Memcached.	memcached.ping, memcached.stats	
Modbus	Odczytuje dane Modbus.	modbus.get	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
MQTT	Odbiera opublikowane wartości tematów MQTT.	mqtt.get	Aby skonfigurować szyfrowane połączenie z brokerem MQTT, określ parametry TLS w pliku konfiguracji agenta jako parametry nazwanej sesji lub domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji.
MySQL	Monitorowanie MySQL i jego odmian.	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	Aby skonfigurować szyfrowane połączenie z bazą danych, określ parametry TLS w pliku konfiguracji agenta jako parametry nazwanej sesji lub domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji.
Netlf	Monitorowanie interfejsów sieciowych.	net.if.collisions, net.if.discovery, net.if.get, net.if.in, net.if.out, net.if.total	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Oracle	Monitorowanie 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	Przed użyciem wtyczki zainstaluj Oracle Instant Client.
Proc	Procent wykorzystania CPU przez proces.	proc.cpu.util	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Uwagi
Redis	Monitorowanie serwera Redis.	redis.config, redis.info, redis.ping, redis.slowlog.count	Aby skonfigurować szyfrowane połączenie z Redis, określ parametry TLS w pliku konfiguracji agenta jako parametry nazwanej sesji lub domyślne. Parametrów TLS nie można przekazywać jako parametrów klucza pozycji. Pamiętaj, że nieprawidłowa lub w inny sposób błędna konfiguracja TLS może uniemożliwić uruchomienie Zabbix agent 2, dlatego przed włączeniem TLS sprawdź pliki certyfikatów, uprawnienia i ścieżki.

Smart	Monitorowanie S.M.A.R.T.	smart.attribute.discovery, smart.disk.discovery, smart.disk.get	Minimalna wymagana wersja smartctl to 7.1. Użytkownik uruchamiający Zabbix agent 2 musi mieć uprawnienia sudo/root do smartctl. Wtyczka używa tylko następujących poleceń:
			`/usr/sbin/smartctl -a *`
			`/usr/sbin/smartctl --scan *`
			`/usr/sbin/smartctl -j -V`
			
			Obsługiwane keys mogą być używane z Zabbix agent 2 tylko w systemach Linux/Windows, zarówno jako sprawdzenie pasywne, jak i aktywne.
			Zobacz także:
			Parametry konfiguracji

SW	Lista zainstalowanych pakietów.	system.sw.packages, system.sw.packages.get	Obsługiwane klucze mają te same parametry co klucz Zabbix agent key.
Swap	Rozmiar przestrzeni swap w bajtach/procentach.	system.swap.size	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.
SystemRun	Uruchamia określone polecenie.	system.run	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key. Zobacz także: Parametry konfiguracji wtyczki (Unix/Windows)
Systemd	Monitorowanie usług systemd.	systemd.unit.discovery, systemd.unit.get, systemd.unit.info	
TCP	Sprawdzanie dostępności połączenia TCP.	net.tcp.port	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.
UDP	Monitorowanie dostępności i wydajności usług UDP.	net.udp.service, net.udp.service.perf	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Uname	Pobieranie informacji o systemie.	system.hostname, system.sw.arch, system.uname	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
Uptime	Zbieranie metryk czasu pracy systemu.	system.uptime	Obsługiwany klucz ma te same parametry co klucz Zabbix agent key.
VFSDev	Zbieranie metryk VFS.	vfs.dev.discovery, vfs.dev.get, vfs.dev.read, vfs.dev.write	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
WebCertificate	Monitorowanie certyfikatów TLS/SSL witryn internetowych.	web.certificate.get	
WebPage	Monitorowanie stron internetowych.	web.page.get, web.page.perf, web.page.regexp	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Uwagi
ZabbixAsync	Asynchroniczne zbieranie metryk.	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	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
ZabbixStats	Wewnętrzne metryki serwera/proxy Zabbix lub liczba opóźnionych pozycji w kolejce.	zabbix.stats	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.
ZabbixSync	Synchroniczne zbieranie metryk.	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.	Obsługiwane klucze mają te same parametry co klucze Zabbix agent keys.

Wtyczki ładowalne

Wtyczki ładowalne nie są dostępne od razu dla Zabbix agent 2 i należy je zainstalować osobno:

- W systemie Linux można użyć pakietów (np. zainstalować Ember+ w systemie Ubuntu za pomocą polecenia `apt install zabbix-agent2-plugin-ember-plus`) lub zbudować wtyczki
- W systemie Windows można zainstalować je z pakietu MSI lub zbudować wtyczki

Attention:
Przed zainstalowaniem wtyczki sprawdź jej plik README. Może on zawierać określone wymagania i instrukcje instalacji.

Kliknij nazwę wtyczki, aby przejść do repozytorium wtyczki, które zawiera jej plik README z dodatkowymi informacjami.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Komentarze
Ceph	Monitorowanie Ceph.	ceph.df.details, ceph.osd.stats, ceph.osd.discovery, ceph.osd.dump, ceph.ping, ceph.pool.discovery, ceph.status	Zobacz także parametry konfiguracji wtyczki Ceph.
Ember+	Monitorowanie Ember+.	ember.get	Zobacz także parametry konfiguracji wtyczki Ember+.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Komentarze
MongoDB	Monitorowanie serwerów i klastrów MongoDB (rozproszona baza danych oparta na dokumentach).	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	Aby skonfigurować szyfrowane połączenia z bazą danych, określ parametry TLS w pliku konfiguracyjnym agenta jako parametry nazwanej sesji. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji. Zobacz także parametry konfiguracji wtyczki MongoDB.
MSSQL	Monitorowanie bazy danych 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	Aby skonfigurować szyfrowane połączenie z bazą danych, określ parametry TLS w pliku konfiguracyjnym agenta jako parametry nazwanej sesji lub domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji. Zobacz także parametry konfiguracji wtyczki MSSQL.
NVIDIA GPU	Monitorowanie procesorów graficznych 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	Zobacz także parametry konfiguracji wtyczki NVIDIA GPU.

Nazwa wtyczki	Opis	Obsługiwane klucze pozycji	Komentarze
PostgreSQL	Monitorowanie PostgreSQL i jego forków.	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	Aby skonfigurować szyfrowane połączenia z bazą danych, określ parametry TLS w pliku konfiguracyjnym agenta jako parametry nazwanej sesji lub domyślne. Obecnie parametrów TLS nie można przekazywać jako parametrów klucza pozycji. Zobacz także parametry konfiguracji wtyczki PostgreSQL.

Note:
Wtyczki ładowalne uruchamiane z użyciem:
 - `-V --version` - wyświetlają informacje o wersji wtyczki i licencji;
 - `-h --help` - wyświetlają informacje pomocy.
 - `-t, --test <item key>` - uruchamiają wtyczkę w celu przeprowadzenia testu (konfiguracja wtyczki jest ignorowana).

Parametry konfiguracji: Zabbix agent 2

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania agent 2 Zabbix.

agent 2 Zabbix to nowa generacja agent Zabbix i może być używany zamiast agent Zabbix.

Określ wartości parametrów w pliku konfiguracyjnym agent 2 Zabbix (`zabbix_agent2.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
Alias	Ustaw alias dla klucza pozycji.
AllowKey	Zezwól na wykonywanie kluczy pozycji pasujących do wzorca.
AllowKeyRegexp	Zezwól na wykonywanie kluczy pozycji pasujących do wzorca opartego na wyrażeniu regularnym.
BufferSend	Ustaw częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix.
BufferSize	Ustaw maksymalną liczbę wartości w buforze pamięci.
ControlSocket	Określ gniazdo sterujące do wysyłania poleceń wykonawczych za pomocą opcji '-R'.
DebugLevel	Ustaw poziom debugowania.
DenyKey	Zabroń wykonywania kluczy pozycji pasujących do wzorca.
DenyKeyRegexp	Zabroń wykonywania kluczy pozycji pasujących do wzorca opartego na wyrażeniu regularnym.
EnablePersistentBuffer	Włącz używanie lokalnego trwałego magazynu dla aktywnych pozycji.
EnableProfiler	Włącz okresowe profilowanie Go podczas uruchamiania agent.

Parametr	Opis
ForceActiveChecksOnStart	Wykonuj aktywne sprawdzenia natychmiast po ponownym uruchomieniu dla pierwszej odebranej konfiguracji.
HeartbeatFrequency	Ustaw częstotliwość komunikatów heartbeat w sekundach.
HostInterface	Określ interfejs host.
HostInterfaceItem	Określ pozycję służącą do pobierania interfejsu host.
HostMetadata	Określ metadane host.
HostMetadataItem	Określ pozycję agent Zabbix służącą do pobierania metadanych host.
Hostname	Określ nazwę host.
HostnameItem	Określ pozycję agent Zabbix służącą do pobierania nazwy host.
Include	Określ katalog lub pojedyncze pliki, które mają zostać uwzględnione w pliku konfiguracyjnym.
ListenIP	Określ adres IP nasłuchiwania agent dla połączeń z serwera/proxy.
ListenPort	Określ port nasłuchiwania agent.
LogFile	Określ plik dziennika agent.
LogFileSize	Ustaw maksymalny rozmiar pliku dziennika.
LogType	Określ typ wyjścia dziennika.
PersistentBufferFile	Określ plik przechowujący bazę danych SQLite.
PersistentBufferPeriod	Ustaw okres przechowywania danych w przypadku braku połączenia z serwerem lub proxy Zabbix.
PidFile	Określ plik PID.
Plugins.<PluginName>.SystemCapacity	Ustaw limit sprawdzeń dla każdej wtyczki, które mogą być wykonywane jednocześnie.
Plugins.Log.MaxLinesPerSecond	Ustaw maksymalną liczbę nowych wierszy na sekundę wysyłanych do serwera/proxy Zabbix podczas przetwarzania aktywnych sprawdzeń `log` lub `logrt`.
Plugins.SystemRun.LogRemoteCommands	Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
PluginSocket	Określ ścieżkę do gniazda UNIX używanego do komunikacji z ładowalnymi wtyczkami.
PluginTimeout	Ustaw limit czasu dla połączeń z ładowalnymi wtyczkami.
ProfilerDir	Katalog, w którym będą zapisywane dane profilowania.
ProfilerMaxFilesPerProfile	Maksymalna liczba plików przechowywanych dla każdego typu profilu.
ProfilerInterval	Interwał zapisu danych profilowania.
RefreshActiveChecks	Określ częstotliwość odświeżania listy aktywnych sprawdzeń.
Server	Określ adresy IP lub nazwy DNS serwerów/proxy Zabbix.
ServerActive	Określ adres serwera/proxy Zabbix lub konfigurację klastra, z której mają być pobierane aktywne sprawdzenia.
SourceIP	Określ źródłowy adres IP.
StatusPort	Określ port nasłuchiwania żądań stanu HTTP (`http://localhost:<port>/status`).
Timeout	Określ, jak długo (w sekundach) oczekiwać na ustanowienie połączenia i wymianę danych z serwerem lub proxy Zabbix.
TLSAccept	Określ wymagany poziom szyfrowania dla połączeń przychodzących.
TLSCAFile	Określ pełną nazwę ścieżki pliku zawierającego certyfikaty najwyższego poziomu CA do weryfikacji certyfikatu peera, używane w szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Określ pełną nazwę ścieżki pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll	Określ ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherAll13	Określ ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert	Określ ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherCert13	Określ ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK	Określ ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSCipherPSK13	Określ ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Określ wymagany poziom szyfrowania dla połączeń z serwerem lub proxy Zabbix.
TLSCRLFile	Określ pełną nazwę ścieżki pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Określ pełną nazwę ścieżki pliku zawierającego klucz prywatny agent, używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Określ pełną nazwę ścieżki pliku zawierającego klucz wstępnie współdzielony agent, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSPSKIdentity	Określ ciąg identyfikatora klucza wstępnie współdzielonego, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.

Parametr	Opis
TLSServerCertIssuer	Określ dozwolonego wystawcę certyfikatu serwera (proxy).
TLSServerCertSubject	Określ dozwolony podmiot certyfikatu serwera (proxy).
UnsafeUserParameters	Zezwól na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
UserParameter	Określ parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Określ domyślną ścieżkę wyszukiwania poleceń UserParameter.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne procesu, a nie wartości znajdujące się w dostarczanych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8, bez BOM.
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Alias

Ustaw alias dla klucza pozycji. Można go użyć do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów `Alias`. Nie można używać wielu parametrów z takim samym kluczem `Alias`.
 Różne klucze `Alias` mogą odwoływać się do tego samego klucza pozycji.
 Aliasów można używać w parametrze `HostMetadataItem`, ale nie w parametrze `HostnameItem`.

Przykład 1: Pobieranie identyfikatora użytkownika `zabbix`.

```
Alias=zabbix.userid:vfs.file.regexp[/etc/passwd,"^zabbix:.:([0-9]+)",,,,\1]
```

Teraz można użyć skróconego klucza `zabbix.userid` do pobierania danych.

Przykład 2: Pobieranie wykorzystania procesora z parametrami domyślnymi i niestandardowymi.

```
Alias=cpu.util:system.cpu.util
Alias=cpu.util[*]:system.cpu.util[*]
```

Umożliwia to użycie klucza `cpu.util` do pobierania procentowego wykorzystania procesora z parametrami domyślnymi, a także użycie `cpu.util[all, idle, avg15]` do pobierania określonych danych dotyczących wykorzystania procesora.

Przykład 3: Uruchamianie wielu reguł niskopoziomowego wykrywania przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz można skonfigurować kilka reguł wykrywania przy użyciu `vfs.fs.discovery`, z różnymi parametrami dla każdej reguły, np. `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym, które obsługuje znak * do dopasowania dowolnej liczby dowolnych znaków.
Wiele reguł dopasowywania kluczy może być zdefiniowanych w połączeniu z `DenyKey`. Wiele reguł dopasowywania kluczy może być zdefiniowanych w połączeniu z `DenyKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Kolejność określa, która z pasujących reguł będzie miała priorytet. Zobacz także: Ograniczanie sprawdzeń agent.

AllowKeyRegexp

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `DenyKeyRegexp`, `AllowKey` i `DenyKey`. Parametry są przetwarzane po kolei, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie kontroli agent.

Przykład:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

BufferSend

Ustaw częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix (w sekundach).
 Należy pamiętać, że jeśli bufor jest pełny, dane zostaną wysłane wcześniej.

Domyślnie: 5
 Zakres: 1-3600

BufferSize

Ustaw maksymalną liczbę wartości w buforze pamięci.
 Agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor zostanie zapełniony. Tego parametru należy używać tylko wtedy, gdy trwały bufor jest wyłączony (`EnablePersistentBuffer=0`).

Domyślnie: 1000
 Zakres: 2-65535

ControlSocket

Określ gniazdo sterujące do wysyłania poleceń w czasie działania za pomocą opcji `-R`.

Domyślnie: `/tmp/agent.sock`

DebugLevel

Ustaw poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca.
 Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak `*`, który dopasowuje dowolną liczbę dowolnych znaków.
Można definiować wiele reguł dopasowywania kluczy w połączeniu z `AllowKey`. Można definiować wiele reguł dopasowywania kluczy w połączeniu z `AllowKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie kontroli agent.

Wartość domyślna: `DenyKey=system.run[*]`

DenyKeyRegexp

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `AllowKeyRegexp`, `AllowKey` i `DenyKey`. Parametry są przetwarzane po kolei, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo.
 Wszystkie pozycje `system.run[]` są domyślnie zabronione (`DenyKey=system.run[*]`), chyba że określono inną regułę `system.run[*]`.

Zobacz także: Ograniczanie kontroli agenta.

Przykład:

```
DenyKeyRegexp=^system\.run\[.*(?:rm -rf|poweroff|reboot).*\]$
```

EnablePersistentBuffer

Włącz użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji.
 Jeśli trwała pamięć masowa jest wyłączona, używany będzie bufor pamięci.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

EnableProfiler

Włącz okresowe profilowanie Go podczas uruchamiania agent.

Profilowanie można również włączać lub wyłączać w czasie działania za pomocą opcji `-R`. Podczas gdy `periodic_prof_execute` zbiera profile, wszystkie polecenia profilera w czasie działania są odrzucane z powodu zajętości.

Wartość domyślna: 0
 Wartości: 0 - wyłączone, 1 - włączone

ForceActiveChecksOnStart

Wykonuj aktywne kontrole natychmiast po restarcie, dla pierwszej otrzymanej konfiguracji.
 Dostępne również jako parametr konfiguracji dla poszczególnych wtyczek, na przykład: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

HeartbeatFrequency

Ustaw częstotliwość komunikatów heartbeat w sekundach.
 Używane do monitorowania dostępności aktywnych kontroli.
0 - komunikaty heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Określa interfejs hosta (adres IP lub nazwa DNS) używany podczas autorejestracji hosta.
 Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawne skonfigurowanie adresu IP lub DNS.
 Więcej informacji można znaleźć w sekcji Using DNS as default interface.

Jeśli nie zostanie zdefiniowana, wartość zostanie pobrana z `HostInterfaceItem`.

Agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Określ pozycję, która będzie używana do ustalenia interfejsu hosta (adresu IP lub nazwy DNS) podczas automatycznej rejestracji hosta.
 Ta wartość jest używana tylko wtedy, gdy parametr `HostInterface` nie został zdefiniowany.
 Więcej informacji można znaleźć w sekcji Używanie DNS jako domyślnego interfejsu.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja `system.run[]` jest obsługiwana niezależnie od ustawień `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

HostMetadata

Określ metadane używane do identyfikowania lub rozróżniania hosta podczas automatycznej rejestracji hosta.
 `HostMetadata` umożliwia rozróżnianie hostów niezależnie od ich nazw.

Jeśli wartość nie zostanie zdefiniowana, zostanie pobrana z `HostMetadataItem`.

agent zgłosi błąd i nie uruchomi się, jeśli określona wartość przekroczy limit 2034 bajtów lub będzie ciągiem znaków innym niż UTF-8. Jeśli parametr oczekuje adresu IP lub nazwy DNS, wartości poprawne w UTF-8, ale niebędące prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Zakres: 0–2034 bajtów

HostMetadataItem

Określa pozycję używaną do uzyskania metadanych hosta.
 Ta opcja jest używana tylko wtedy, gdy parametr `HostMetadata` nie został zdefiniowany.

Wartość `HostMetadataItem` jest pobierana przy każdej próbie automatycznej rejestracji i używana wyłącznie podczas procesu automatycznej rejestracji hosta. `HostMetadataItem` umożliwia rozróżnianie hostów niezależnie od ich nazw.

Obsługiwane są parametry użytkownika i aliasy. Pozycja `system.run[]` jest obsługiwana niezależnie od ustawień `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy w dzienniku, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 65535 punktów kodowych UTF-8. Wartość zwracana przez pozycję musi być ciągiem UTF-8; w przeciwnym razie zostanie zignorowana. Jeśli parametr oczekuje adresu IP lub nazwy DNS, wartości, które są prawidłowym tekstem UTF-8, ale nie są prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Nazwa hosta

Określ unikalne nazwy hostów rozróżniające wielkość liter. Lista rozdzielona przecinkami.
 Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze.
 Wartość jest pobierana z `HostnameItem`, jeśli nie została zdefiniowana.
 Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na nazwę hosta, 2048 znaków dla całej linii.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Określ pozycję służącą do pobierania nazwy hosta.
 Ta opcja jest używana tylko wtedy, gdy `Hostname` nie jest zdefiniowany.
 Parametry użytkownika ani aliasy nie są obsługiwane, ale pozycja `system.run[]` jest obsługiwana niezależnie od wartości `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

Wartość domyślna: `system.hostname`

Include

Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracyjnego.
 Podczas instalacji Zabbix ten parametr jest ustawiany na `/usr/local/etc`, chyba że zmienisz to podczas kompilacji.

Ogólne zasady dla parametru `Include`:

- Ścieżka może być względna względem lokalizacji pliku zabbix_agent2.conf.
- W jednym pliku konfiguracyjnym można określić wiele parametrów Include.
- Zabbix przetwarza parametr Include natychmiast po jego odczytaniu, więc warto umieścić go na końcu pliku konfiguracyjnego.

Zasady dotyczące plików dołączanych:

- Każdy plik musi być możliwy do odczytu.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie zostanie uruchomiony.

Zasady dotyczące dołączania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności dołączania (np. pliki nie są dołączane alfabetycznie).
- Nie definiuj tego samego parametru w więcej niż jednym dołączonym pliku (np. aby nadpisać ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Zwracaj uwagę na kopie zapasowe tworzone automatycznie przez niektóre edytory tekstu (np. edycja include/my_specific.conf może utworzyć include/my_specific.conf.BAK), ponieważ oba pliki zostaną dołączone.

Aby dołączyć tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (*) do dopasowywania wzorców.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenIP

Określa adres IP nasłuchiwania agent dla połączeń z serwera/proxy. Lista rozdzielana przecinkami.
 Pierwszy adres IP jest wysyłany do serwera Zabbix, jeśli następuje połączenie z nim, w celu pobrania listy aktywnych kontroli.

Domyślnie: 0.0.0.0

ListenPort

Określa port nasłuchu agenta dla połączeń z serwera/proxy.

Domyślnie: 10050
 Zakres: 1024-32767

LogFile

Określ plik dziennika agenta.

Domyślnie: /tmp/zabbix_agent2.log
 Wymagane: Tak, jeśli LogType jest ustawione na file; w przeciwnym razie nie

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
0 - wyłącza automatyczną rotację dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika i rotacja pliku z jakiegokolwiek powodu się nie powiedzie, istniejący plik dziennika zostanie skrócony i utworzony od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogType

Określ typ wyjścia logów: - *file* - zapisuj log do pliku określonego parametrem LogFile; - *system* - zapisuj log do sysloga; - *console* - zapisuj log do standardowego wyjścia.

Domyślnie: file

PersistentBufferFile

Określ plik do przechowywania bazy danych SQLite. Musi to być pełna nazwa pliku.
 Ten parametr jest używany tylko wtedy, gdy włączony jest trwały bufor (EnablePersistentBuffer=1).

PersistentBufferPeriod

Ustaw okres przechowywania danych, gdy nie ma połączenia z serwerem Zabbix lub proxy. Starsze dane zostaną utracone.
 Dane dziennika będą zachowane. Ten parametr jest używany tylko wtedy, gdy włączony jest trwały bufor (EnablePersistentBuffer=1).

Domyślnie: 1h
 Zakres: 1m-365d

PidFile

Określa plik PID.

Domyślnie: /tmp/zabbix_agent2.pid

Wtyczki.<PluginName>.System.Capacity

Ustaw limit jednoczesnych sprawdzeń dla wtyczki <PluginName>, które mogą być wykonywane w tym samym czasie.

Domyślnie: 1000 Zakres: 1-1000

Plugins.Log.MaxLinesPerSecond

Ustaw maksymalną liczbę nowych linii na sekundę wysyłanych do serwera/proxy Zabbix podczas przetwarzania aktywnych kontroli log lub logrt.
 Podana wartość zostanie zastąpiona przez parametr maxlines, określony w kluczu pozycja log i logrt.
Uwaga: Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w MaxLinesPerSecond, aby wyszukać wymagany ciąg w pozycjach dziennika.

Domyślnie: 20
 Zakres: 1-1000

Plugins.SystemRun.LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
 Polecenia będą rejestrowane tylko wtedy, gdy zostaną wykonane zdalnie.
 Wpisy dziennika nie będą tworzone, jeśli system.run[] zostanie uruchomione lokalnie przez parametry HostMetadataItem, HostInterfaceItem lub HostnameItem.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

PluginSocket

Określ ścieżkę do gniazda UNIX dla komunikacji z ładowalnymi wtyczkami.

Domyślnie: /tmp/agent.plugin.sock

PluginTimeout

Określa limit czasu dla połączeń z wczytywalnymi wtyczkami, w sekundach.

Domyślnie: Timeout
 Zakres: 1-30

ProfilerDir

Katalog, w którym będą zapisywane dane profilowania.

Wartość domyślna: /tmp/zabbix_agent2_profiles

ProfilerMaxFilesPerProfile

Maksymalna liczba plików przechowywanych dla każdego typu profilu. Limit jest stosowany do ukończonych plików po każdym zrzucie profilu. Aktywny profil CPU jest zbierany w ukrytym pliku tymczasowym i nie jest uwzględniany w limicie. Pliki utworzone przez periodic_prof_execute również podlegają temu limitowi.

Wartość domyślna: 10
 Zakres: 1-100

ProfilerInterval

Interwał zapisu danych profilowania w sekundach.

Wartość domyślna: 3600
 Zakres: 1-86400

RefreshActiveChecks

Określ częstotliwość odświeżania listy aktywnych kontroli, w sekundach.
 Należy pamiętać, że po nieudanym odświeżeniu aktywnych kontroli następna próba odświeżenia zostanie podjęta po 60 sekundach.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Określ adresy IP, opcjonalnie w notacji CIDR, lub nazwy DNS serwerów/proxy Zabbix. Lista rozdzielana przecinkami.
 Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, 127.0.0.1, ::127.0.0.1, ::ffff:127.0.0.1 są traktowane jednakowo, a ::/0 zezwoli na dowolny adres IPv4 lub IPv6. 0.0.0.0/0 można użyć, aby zezwolić na dowolny adres IPv4. Dozwolone są spacje. Jeśli ten parametr nie zostanie określony, wyłącza to pasywne sprawdzanie, a agent nie będzie nasłuchiwał na żadnym porcie TCP.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Określa adres serwera/proxy Zabbix lub konfigurację klastra, z którego mają być pobierane aktywne kontrole.
 Adres serwera/proxy jest adresem IP lub nazwą DNS, a opcjonalny port jest oddzielony dwukropkiem.
Konfiguracja klastra to jeden lub więcej adresów członków grupy serwera lub proxy, oddzielonych średnikiem. Można podać wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami. Jeśli nie są używane grupy proxy, nie należy podawać więcej niż jednego

proxy Zabbix dla każdego serwera/klastra Zabbix. Jeśli podano proxy Zabbix, nie należy podawać serwera/klastra Zabbix dla tego proxy.
Można podać wiele adresów rozdzielonych przecinkami, aby używać kilku niezależnych serwerów Zabbix równolegle. Dozwolone są spacje.
Jeśli port nie zostanie określony, używany jest port domyślny.
Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla tego hosta określono port. Jeśli port nie jest określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.
Jeśli ten parametr nie zostanie określony, aktywne kontrole są wyłączone.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,[::1]:30051,::1,[12fc::1]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Określ źródłowy adres IP dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix.
- nawiązywania połączeń podczas wykonywania niektórych pozycji (`web.page.get`, `net.tcp.port` itp.).

StatusPort

Określ port nasłuchiwania dla żądań statusu HTTP (`http://localhost:<port>/status`).

Zakres: 1024-32767

Limit czasu

Określa, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy Zabbix lub serwerem.

Ten parametr definiuje czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwanie na odpowiedź z serwera Zabbix;
- wysyłanie żądań do serwera Zabbix, w tym żądań konfiguracji pozycji i danych pozycji w active checks;
- pobieranie danych dziennika przez logfile;
- wysyłanie komunikatów heartbeat;
- maksymalny czas trwania sprawdzeń `vfs.*`;
- pełnienie roli wartości awaryjnej w scenariuszach, gdy serwer lub proxy w wersji starszej niż 7.0 wysyła sprawdzenia bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych sprawdzeń agent, które mają konfigurowalne ustawienia limitu czasu we frontendzie (na poziomie globalnym, proxy lub dla pojedynczej pozycji).

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Określ poziom szyfrowania wymagany dla połączeń przychodzących.
 Używane dla pasywnych kontroli.
 Można podać wiele wartości, oddzielając je przecinkiem: - *unencrypted* - akceptuj połączenia bez szyfrowania (domyślnie); - *psk* - akceptuj połączenia z TLS i kluczem wstępnie współdzielonym (PSK); - *cert* - akceptuj połączenia z TLS i certyfikatem.

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Podaj pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używaną w szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów, używany do szyfrowanej komunikacji z komponentami Zabbix.

TLSCipherAll

Określ ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128:kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherAll13

Określ ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert

Określ ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Przykład:

```
EECDH+aRSA+AES128:RSA+aRSA+AES128
```

Należy pamiętać, że tego parametru nie można używać razem z `TLSAccept=cert,psk`; dla połączenia certyfikatem (`TLSConnect=cert`) użyj zamiast tego `TLSCipherAll`.

TLSCipherCert13

Określ ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Należy pamiętać, że tego parametru nie można używać razem z `TLSAccept=cert,psk`; dla połączenia certyfikatowego (`TLSConnect=cert`) użyj zamiast tego `TLSCipherAll13`.

TLSCipherPSK

Określ ciąg szyfrów OpenSSL (TLS 1.2). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
kECDHEPSK+AES128:kPSK+AES128
```

TLSCipherPSK13

Określ ciąg szyfrów OpenSSL (TLS 1.3) dla OpenSSL 1.1.1 lub nowszego. Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Określ poziom szyfrowania wymagany dla połączeń agent z serwerem Zabbix lub proxy.
 Używane dla aktywnych kontroli.
 Można określić tylko jedną wartość: - *unencrypted* - połączenie bez szyfrowania (domyślnie); - *psk* - połączenie z użyciem TLS i współdzielonego klucza wstępnego (PSK); - *cert* - połączenie z użyciem TLS i certyfikatu.

Obowiązkowe: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz agenta, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Określ pełną ścieżkę do pliku zawierającego współdzielony klucz agenta, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSPSKIdentity

Określ ciąg tożsamości pre-shared key, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSServerCertIssuer

Określ dozwolonego wystawcę certyfikatu serwera (proxy).

TLSServerCertSubject

Określ dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Dodatkowo znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

UserParameter

Określ zdefiniowany przez użytkownika parametr do monitorowania. Może istnieć kilka takich parametrów.
Format: `UserParameter=<key>,<shell command>`
Pamiętaj, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie EOL. Polecenia powłoki mogą używać ścieżek względnych, jeśli określono parametr `UserParameterDir`.

Przykład:

```
UserParameter=system.test,who|wc -l
UserParameter=check_cpu,./custom_script.sh
```

UserParameterDir

Określa domyślną ścieżkę wyszukiwania dla poleceń `UserParameter`. Jeśli jest używana, agent zmieni swój bieżący katalog roboczy na katalog wskazany tutaj przed wykonaniem polecenia. Dzięki temu polecenia UserParameter mogą używać względnego prefiksu `./` zamiast pełnej ścieżki.
Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=/opt/myscripts
```

Parametry konfiguracji: Zabbix agent 2 (Windows)

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania Zabbix agent 2 w systemie Windows.

Określ wartości parametrów w pliku konfiguracyjnym agent 2 Zabbix (`zabbix_agent2.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
Alias	Ustawia alias dla klucza pozycji.
AllowKey	Zezwala na wykonywanie kluczy pozycji pasujących do wzorca.
AllowKeyRegexp	Zezwala na wykonywanie kluczy pozycji pasujących do wzorca opartego na wyrażeniu regularnym.
BufferSend	Ustawia częstotliwość wysyłania wartości z bufora do serwera Zabbix.
BufferSize	Ustawia maksymalną liczbę wartości w buforze pamięci.
ControlSocket	Określa gniazdo sterujące używane do wysyłania poleceń wykonywanych w czasie działania za pomocą opcji `-R`.
DebugLevel	Ustawia poziom debugowania.
DenyKey	Zabrania wykonywania kluczy pozycji pasujących do wzorca.
DenyKeyRegexp	Zabrania wykonywania kluczy pozycji pasujących do wzorca opartego na wyrażeniu regularnym.
EnablePersistentBuffer	Włącza używanie lokalnego trwałego magazynu dla aktywnych pozycji.
EnableProfiler	Włącza okresowe profilowanie Go podczas uruchamiania agent.
ForceActiveChecksOnStart	Wykonuje aktywne sprawdzenia natychmiast po ponownym uruchomieniu dla pierwszej odebranej konfiguracji.
HeartbeatFrequency	Ustawia częstotliwość komunikatów heartbeat w sekundach.

Parametr	Opis
HostInterface	Określa interfejs hosta.
HostInterfaceItem	Określa pozycję używaną do pobierania interfejsu hosta.
HostMetadata	Określa metadane hosta.
HostMetadataItem	Określa pozycję agent Zabbix używaną do pobierania metadanych hosta.
Hostname	Określa nazwę hosta.
HostnameItem	Określa pozycję agent Zabbix używaną do pobierania nazwy hosta.
Include	Określa katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracyjnego.
ListenIP	Określa adres IP nasłuchiwania agent dla połączeń z serwera/proxy.
ListenPort	Określa port nasłuchiwania agent.
LogFile	Określa plik dziennika agent.
LogFileSize	Ustawia maksymalny rozmiar pliku dziennika.
LogType	Określa typ wyjścia dziennika.
PerfCounter	Określa nowy parametr <parameter_name>, będący średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach).
PerfCounterEn	Określa nowy parametr <parameter_name>, będący średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego okresu <period> (w sekundach). W odróżnieniu od PerfCounter ścieżki liczników wydajności muszą być podane w języku angielskim.
PersistentBufferFile	Określa plik, w którym Zabbix agent 2 powinien przechowywać bazę danych SQLite.
PersistentBufferPeriod	Ustawia czas przechowywania danych w przypadku braku połączenia z serwerem lub proxy.
Plugins.<PluginName>.SystemCapacity	Ustawia limit sprawdzeń dla każdej wtyczki, które mogą być wykonywane jednocześnie.
Plugins.Log.MaxLinesPerSecond	Ustawia maksymalną liczbę nowych wierszy, które agent wyśle na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych sprawdzeń `log`, `logrt` i `eventlog`.
Plugins.SystemRun.LogRemoteCommands	Włącza rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń.
PluginSocket	Określa ścieżkę do nazwanej potoki systemu Windows używanej do komunikacji z ładowalnymi wtyczkami.
PluginTimeout	Ustawia limit czasu dla połączeń z ładowalnymi wtyczkami, w sekundach.
ProfilerDir	Katalog, w którym będą zapisywane dane profilowania.
ProfilerMaxFilesPerProfile	Maksymalna liczba plików przechowywanych dla każdego typu profilu.
ProfilerInterval	Interwał zapisu danych profilowania.
RefreshActiveChecks	Określa częstotliwość odświeżania listy aktywnych sprawdzeń.
Server	Określa adresy IP lub nazwy DNS serwerów/proxy Zabbix.
ServerActive	Określa adres serwera/proxy Zabbix lub konfigurację klastra, z której mają być pobierane aktywne sprawdzenia.
SourceIP	Określa źródłowy adres IP.
StatusPort	Ustawia port, na którym agent nasłuchuje żądań HTTP dotyczących stanu (`http://localhost:<port>/status`).
Timeout	Określa czas oczekiwania (w sekundach) na ustanowienie połączenia i wymianę danych z serwerem lub proxy Zabbix.
TLSAccept	Określa wymagany poziom szyfrowania dla połączeń przychodzących.
TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) używane do weryfikacji certyfikatu peera w szyfrowanej komunikacji między komponentami Zabbix.
TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSCipherAll13	Określa ciąg szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.
TLSCipherCert13	Określa ciąg szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.
TLSCipherPSK13	Określa ciąg szyfrów OpenSSL (TLS 1.3). Zastępuje domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.
TLSConnect	Określa wymagany poziom szyfrowania dla połączeń z serwerem lub proxy Zabbix.
TLSCRLFile	Określa pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.
TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego klucz prywatny agent używany w szyfrowanej komunikacji między komponentami Zabbix.
TLSPSKFile	Określa pełną ścieżkę do pliku zawierającego klucz wstępnie współdzielony agent, używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSPSKIdentity	Określa ciąg identyfikatora klucza wstępnie współdzielonego używany w szyfrowanej komunikacji z serwerem lub proxy Zabbix.
TLSServerCertIssuer	Określa dozwolonego wystawcę certyfikatu serwera (proxy).
TLSServerCertSubject	Określa dozwolony podmiot certyfikatu serwera (proxy).

Parametr	Opis
UnsafeUserParameters	Zezwala na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika.
UserParameter	Określa parametr zdefiniowany przez użytkownika do monitorowania.
UserParameterDir	Określa domyślną ścieżkę wyszukiwania poleceń `UserParameter`.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne procesu, a nie wartości znajdujące się w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8, bez BOM;
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Alias

Ustaw alias dla klucza pozycji. Można go użyć do zastąpienia długiego i złożonego klucza pozycji krótszym i prostszym.
 Może występować wiele parametrów `Alias`. Nie można używać wielu parametrów z takim samym kluczem `Alias`.
 Różne klucze `Alias` mogą odwoływać się do tego samego klucza pozycji.
 Aliasów można używać w parametrze `HostMetadataItem`, ale nie w parametrze `HostnameItem`.

Przykład 1: Pobieranie procentowego użycia pliku stronicowania z serwera.

```
Alias=pg_usage:perf_counter[\Paging File(_Total)\% Usage]
```

Teraz można użyć skróconego klucza `pg_usage` do pobierania danych.

Przykład 2: Pobieranie obciążenia procesora z użyciem parametrów domyślnych i niestandardowych.

```
Alias=cpu.load:system.cpu.load
Alias=cpu.load[*]:system.cpu.load[*]
```

Umożliwia to użycie klucza `cpu.load` do pobierania obciążenia procesora z parametrami domyślnymi, a także użycie `cpu.load[percpu,avg15]` do pobierania szczegółowych danych dotyczących obciążenia procesora.

Przykład 3: Uruchamianie wielu reguł wykrywania niskopoziomowego przetwarzających te same pozycje wykrywania.

```
Alias=vfs.fs.discovery[*]:vfs.fs.discovery
```

Teraz można skonfigurować kilka reguł wykrywania przy użyciu `vfs.fs.discovery` z różnymi parametrami dla każdej reguły, np. `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` itd.

AllowKey

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym, które obsługuje znak `*` do dopasowania dowolnej liczby dowolnych znaków.
 Wiele reguł dopasowywania kluczy może być zdefiniowanych w połączeniu z `DenyKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane jeden po drugim zgodnie z kolejnością ich występowania. Kolejność określa, która z pasujących reguł będzie miała priorytet. Zobacz także: Ograniczanie sprawdzeń agent.

AllowKeyRegexp

Zezwalaj na wykonywanie tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `DenyKeyRegexp`, `AllowKey` i `DenyKey`. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo. Zobacz także: Ograniczanie kontroli agent.

Przykład:

```
AllowKeyRegexp=^system\.run\[(?:/usr)?/bin/hostname(?: -[fsd])?\]$
```

BufferSend

Ustaw częstotliwość wysyłania wartości z bufora do serwera/proxy Zabbix (w sekundach).
Uwaga: jeśli bufor jest pełny, dane zostaną wysłane wcześniej.

Domyślnie: 5
 Zakres: 1-3600

BufferSize

Ustaw maksymalną liczbę wartości w buforze pamięci.
 Agent wyśle wszystkie zebrane dane do serwera Zabbix lub proxy, jeśli bufor zostanie zapełniony.
Ten parametr należy używać tylko wtedy, gdy trwały bufor jest wyłączony (`EnablePersistentBuffer=0`).

Domyślnie: 1000
 Zakres: 2-65535

ControlSocket

Określ gniazdo sterujące, używane do wysyłania poleceń w czasie działania za pomocą opcji `-R`.

Domyślnie: `\\.\pipe\agent.sock`

DebugLevel

Ustaw poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

DenyKey

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem wieloznacznym obsługującym znak `*`, który dopasowuje dowolną liczbę dowolnych znaków.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `AllowKey`, `AllowKeyRegexp` i `DenyKeyRegexp`. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo.

Zobacz także: Ograniczanie kontroli agent.

Domyślnie: `DenyKey=system.run[*]`

DenyKeyRegexp

Zabroń wykonywania tych kluczy pozycji, które pasują do wzorca. Wzorzec klucza jest wyrażeniem regularnym.
 Można definiować wiele reguł dopasowywania kluczy w połączeniu z `AllowKeyRegexp`, `AllowKey` i `DenyKey`. Parametry są przetwarzane pojedynczo, zgodnie z kolejnością ich występowania. Kolejność określa, która pasująca reguła będzie miała pierwszeństwo.
 Wszystkie pozycje `system.run[]` są domyślnie zabronione (`DenyKey=system.run[*]`), chyba że określono inną regułę `system.run[*]`.

Zobacz także: Ograniczanie kontroli agent.

EnablePersistentBuffer

Włącz użycie lokalnej trwałej pamięci masowej dla aktywnych pozycji. Jeśli trwała pamięć masowa jest wyłączona, używany będzie bufor pamięci.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

EnableProfiler

Włącz okresowe profilowanie Go podczas uruchamiania agent.

Profilowanie można również włączać lub wyłączać w czasie działania za pomocą opcji `-R`. Podczas gdy `periodic_prof_execute` zbiera profile, wszystkie polecenia profilera w czasie działania są odrzucane z powodu zajętości.

Wartość domyślna: 0
 Wartości: 0 - wyłączone, 1 - włączone

ForceActiveChecksOnStart

Wykonuj aktywne sprawdzenia natychmiast po restarcie, aby pobrać pierwszą otrzymaną konfigurację. Dostępne również jako parametr konfiguracji dla poszczególnych wtyczek, na przykład: `Plugins.Uptime.System.ForceActiveChecksOnStart=1`

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

HeartbeatFrequency

Ustaw częstotliwość wiadomości heartbeat w sekundach. Używane do monitorowania dostępności aktywnych kontroli.
0 - wiadomości heartbeat wyłączone.

Domyślnie: 60
 Zakres: 0-3600

HostInterface

Określa interfejs hosta (adres IP lub nazwa DNS) używany podczas autorejestracji hosta. Ta wartość zostanie użyta do wypełnienia interfejsu w nowo utworzonym hoście i umożliwia jawne skonfigurowanie adresu IP lub DNS. Więcej informacji można znaleźć w sekcji Using DNS as default interface.

Jeśli nie zostanie zdefiniowana, wartość zostanie pobrana z HostInterfaceItem.

agent zgłosi błąd i nie uruchomi się, jeśli wartość przekroczy limit 255 znaków.

Zakres: 0-255 znaków

HostInterfaceItem

Określ pozycję służącą do ustalenia interfejsu hosta (adresu IP lub nazwy DNS) podczas automatycznej rejestracji hosta. Ta wartość jest używana tylko wtedy, gdy parametr `HostInterface` nie został zdefiniowany. Więcej informacji zawiera sekcja Używanie DNS jako domyślnego interfejsu.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 255 znaków.

Pozycja `system.run[]` jest obsługiwana niezależnie od ustawień `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

HostMetadata

Określ metadane używane do identyfikowania lub rozróżniania hosta podczas automatycznej rejestracji hosta (aktywny agent). `HostMetadata` umożliwia rozróżnianie hostów niezależnie od ich nazw.

Jeśli parametr nie zostanie zdefiniowany, wartość zostanie pobrana z `HostMetadataItem`.

agent zgłosi błąd i nie uruchomi się, jeśli określona wartość przekroczy limit 2034 bajtów lub będzie ciągiem znaków innym niż UTF-8.
Jeśli parametr oczekuje adresu IP lub nazwy DNS, wartości poprawne w UTF-8, ale niebędące prawidłowymi adresami IP ani nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Zakres: 0–2034 bajty

HostMetadataItem

Określa pozycję agenta Zabbix, z której mają być pobierane metadane hosta. Ta opcja jest używana tylko wtedy, gdy parametr `HostMetadata` nie został zdefiniowany.

Wartość `HostMetadataItem` jest pobierana przy każdej próbie automatycznej rejestracji i używana wyłącznie podczas procesu automatycznej rejestracji hosta (aktywny agent). `HostMetadataItem` umożliwia rozróżnianie hostów na podstawie innych informacji niż nazwa hosta.

Obsługiwane są parametry użytkownika i aliasy. Pozycja `system.run[]` jest obsługiwana niezależnie od ustawień `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

Podczas żądania automatycznej rejestracji agent zapisze komunikat ostrzegawczy w dzienniku, jeśli wartość zwrócona przez określoną pozycję przekroczy limit 65535 punktów kodowych UTF-8. Wartość zwrócona przez pozycję musi być ciągiem znaków UTF-8; w przeciwnym razie zostanie zignorowana. Jeśli parametr wymaga adresu IP lub nazwy DNS, wartości będące prawidłowym tekstem UTF-8, ale nieprawidłowymi adresami IP lub nazwami DNS, również zostaną odrzucone i zgłoszone jako nieprawidłowe.

Metadane wielowierszowe nie są obsługiwane — dane wyjściowe zostaną obcięte przy pierwszym znaku końca wiersza.

Hostname

Określ unikalne nazwy hostów rozróżniające wielkość liter. Lista rozdzielona przecinkami.
 Wymagane dla aktywnych kontroli i musi odpowiadać nazwom hostów skonfigurowanym na serwerze. Wartość jest pobierana z HostnameItem, jeśli nie została zdefiniowana.
 Dozwolone znaki: alfanumeryczne, '.', ' ', '_' i '-'. Maksymalna długość: 128 znaków na każdą nazwę hosta, 2048 znaków dla całego wiersza.

Domyślnie: Ustawiane przez HostnameItem

HostnameItem

Określ pozycję służącą do pobierania nazwy hosta. Ta opcja jest używana tylko wtedy, gdy `Hostname` nie jest zdefiniowany. Parametry użytkownika ani aliasy nie są obsługiwane, ale pozycja `system.run[]` jest obsługiwana niezależnie od wartości `AllowKey`, `AllowKeyRegexp`, `DenyKey` lub `DenyKeyRegexp`.

Wartość domyślna: `system.hostname`

Include

Określ katalog lub pojedyncze pliki, które mają zostać uwzględnione w pliku konfiguracyjnym (domyślnie znajdującym się w `C:\Program Files\Zabbix Agent 2`, jeśli agent Zabbix został zainstalowany przy użyciu pakietów instalacyjnych Windows MSI; znajdującym się w folderze określonym podczas instalacji, jeśli agent Zabbix został zainstalowany jako archiwum zip). Podczas instalacji Zabbix ten parametr jest ustawiany na `/usr/local/etc`, chyba że zmienisz go podczas kompilacji.

Ogólne zasady dotyczące parametru `Include`:

- Ścieżka może być względna względem lokalizacji pliku zabbix_agent2.conf (np. Include=.\zabbix_agent2.d\plugins.d*.c
- W jednym pliku konfiguracyjnym można określić wiele parametrów Include.
- Zabbix przetwarza parametr Include natychmiast po jego odczytaniu, dlatego rozważ umieszczenie go na końcu pliku konfiguracyjnego.

Zasady dotyczące uwzględnianych plików:

- Każdy plik musi być możliwy do odczytu.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie zostanie uruchomiony.

Zasady dotyczące uwzględniania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności uwzględniania (np. pliki nie są uwzględniane w kolejności alfabetycznej).
- Nie definiuj tego samego parametru w więcej niż jednym uwzględnianym pliku (np. aby zastąpić ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Zwracaj uwagę na kopie zapasowe tworzone automatycznie przez niektóre edytory tekstu (np. edycja include/my_specific.conf może utworzyć include/my_specific.conf.BAK), ponieważ oba pliki zostaną uwzględnione.

Aby uwzględnić tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (*) do dopasowywania wzorca.

Przykład:

```
Include=C:\Program Files\Zabbix Agent2\zabbix_agent2.d\*.conf
```

ListenIP

Określa adres IP, na którym agent nasłuchuje połączeń z serwera/proxy. Lista rozdzielona przecinkami.
 Pierwszy adres IP jest wysyłany do serwera Zabbix, jeśli następuje połączenie z nim, aby pobrać listę aktywnych kontroli.

Domyślnie: 0.0.0.0

ListenPort

Określa port nasłuchu agenta dla połączeń z serwera/proxy.

Domyślnie: 10050
 Zakres: 1024-32767

LogFile

Określ plik dziennika agenta.

Domyślnie: c:\zabbix_agent2.log
 Obowiązkowe: Tak, jeśli LogType jest ustawione na file; w przeciwnym razie nie

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
0 - wyłącza automatyczną rotację dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika, a rotacja pliku z jakiegokolwiek powodu się nie powiedzie, istniejący plik dziennika zostanie obcięty i utworzony od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogType

Określ typ wyjścia logu: - *file* - zapisuj log do pliku określonego parametrem LogFile; - *console* - zapisuj log do standardowego wyjścia.

Domyślnie: file

PerfCounter

Zdefiniuj nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego przedziału czasu <period> (w sekundach).
Składnia: <parameter_name>,”<perf_counter_path>”,<period>

Na przykład, jeśli chcesz otrzymywać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr ”interrupts” w następujący sposób:

```
PerfCounter = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycja podczas tworzenia pozycji. Próbki do obliczania średniej wartości będą pobierane co sekundę.
Możesz uruchomić typeperf -qx, aby uzyskać listę wszystkich liczników wydajności dostępnych w systemie Windows.

PerfCounterEn

Zdefiniuj nowy parametr <parameter_name>, który jest średnią wartością licznika wydajności systemu <perf_counter_path> dla określonego przedziału czasu <period> (w sekundach). W porównaniu z PerfCounter, ścieżki perfcounter muszą być w języku angielskim. Obsługiwane tylko w **Windows Server 2008/Vista** i nowszych.
Składnia: <parameter_name>,”<perf_counter_path>”,<period>

Na przykład, jeśli chcesz otrzymywać średnią liczbę przerwań procesora na sekundę z ostatniej minuty, możesz zdefiniować nowy parametr ”interrupts” w następujący sposób:

```
PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60
```

Zwróć uwagę na podwójne cudzysłowy wokół ścieżki licznika wydajności. Nazwa parametru (interrupts) ma być używana jako klucz pozycji podczas tworzenia pozycji. Próbki do obliczania średniej wartości będą pobierane co sekundę.
Listę angielskich ciągów można znaleźć, przeglądając następujący klucz rejestru: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009.

PersistentBufferFile

Określ plik, w którym agent 2 Zabbix powinien przechowywać bazę danych SQLite. Musi to być pełna nazwa pliku.
 Ten parametr jest używany tylko wtedy, gdy włączony jest trwały bufor (EnablePersistentBuffer=1).

PersistentBufferPeriod

Ustaw czas przechowywania danych, gdy nie ma połączenia z serwerem lub proxy. Starsze dane zostaną utracone. Dane dziennika będą zachowane.
 Ten parametr jest używany tylko wtedy, gdy bufor trwały jest włączony (EnablePersistentBuffer=1).

Domyślnie: 1h
 Zakres: 1m-365d

Wtyczki.<PluginName>.System.Capacity

Ustaw limit kontroli na wtyczkę <PluginName>, które mogą być wykonywane jednocześnie.

Domyślnie: 1000 Zakres: 1-1000

Plugins.Log.MaxLinesPerSecond

Ustaw maksymalną liczbę nowych linii, które agent będzie wysyłał na sekundę do serwera Zabbix lub proxy podczas przetwarzania aktywnych kontroli log, logrt i eventlog. Podana wartość zostanie zastąpiona przez parametr maxlines podany w kluczu pozycja log, logrt lub eventlog.
Uwaga: Zabbix będzie przetwarzać 10 razy więcej nowych linii niż ustawiono w MaxLinesPerSecond, aby wyszukać wymagany ciąg w pozycjach dziennika.

Domyślnie: 20
 Zakres: 1-1000

Plugins.SystemRun.LogRemoteCommands

Włącz rejestrowanie wykonywanych poleceń powłoki jako ostrzeżeń. Polecenia będą rejestrowane tylko wtedy, gdy zostaną wykonane zdalnie. Wpisy dziennika nie będą tworzone, jeśli system.run[] zostanie uruchomione lokalnie przez parametry HostMetadataItem, HostInterfaceItem lub HostnameItem.

Domyślnie: 0
 Wartości: 0 - wyłączone, 1 - włączone

PluginSocket

Określ ścieżkę do nazwanego potoku systemu Windows używanego do komunikacji z ładowalnymi wtyczkami.

Domyślnie: \\.\pipe\agent.plugin.sock

PluginTimeout

Ustaw limit czasu dla połączeń z ładowalnymi wtyczkami, w sekundach.

Domyślnie: Timeout
 Zakres: 1-30

ProfilerDir

Katalog, w którym będą zapisywane dane profilowania.

Wartość domyślna: c:\zabbix_pprof

ProfilerMaxFilesPerProfile

Maksymalna liczba plików przechowywanych dla każdego typu profilu. Limit jest stosowany do ukończonych plików po każdym zrzucie profilu. Aktywny profil procesora jest zbierany w ukrytym pliku tymczasowym i nie jest uwzględniany w limicie. Pliki utworzone przez periodic_prof_execute również podlegają limitowi.

Wartość domyślna: 10
 Zakres: 1-100

ProfilerInterval

Interwał zapisu danych profilowania w sekundach.

Wartość domyślna: 3600
 Zakres: 1-86400

RefreshActiveChecks

Określ częstotliwość odświeżania listy aktywnych kontroli, w sekundach.
 Należy pamiętać, że po nieudanym odświeżeniu aktywnych kontroli następna próba odświeżenia zostanie podjęta po 60 sekundach.

Domyślnie: 5
 Zakres: 1-86400

Serwer

Określ adresy IP, opcjonalnie w notacji CIDR, lub nazwy DNS serwerów/proxy Zabbix. Lista rozdzielana przecinkami.
 Jeśli parametr nie zostanie określony, sprawdzanie pasywne będzie wyłączone, a agent nie będzie nasłuchiwał na żadnym porcie TCP.
 Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4. Dozwolone są spacje.

Przykład:

```
Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

ServerActive

Określa adres serwera/proxy Zabbix lub konfigurację klastra, z którego mają być pobierane aktywne kontrole.
 Jeśli parametr nie zostanie określony, aktywne kontrole są wyłączone.

Adres serwera/proxy jest adresem IP lub nazwą DNS oraz opcjonalnym portem oddzielonym dwukropkiem. Jeśli port nie zostanie określony, używany jest port domyślny.
 Adresy IPv6 muszą być ujęte w nawiasy kwadratowe, jeśli dla tego hosta określono port. Jeśli port nie jest określony, nawiasy kwadratowe dla adresów IPv6 są opcjonalne.

Konfiguracja klastra to jeden lub więcej adresów członków grupy serwera lub proxy oddzielonych średnikiem.

Można określić wiele serwerów/klastrów Zabbix oraz proxy Zabbix, oddzielając je przecinkami.

O ile nie są używane grupy proxy, nie należy określać więcej niż jednego proxy Zabbix z każdego serwera/klastra Zabbix. Jeśli określono proxy Zabbix, nie należy określać serwera/klastra Zabbix dla tego proxy.

Można podać wiele adresów rozdzielonych przecinkami, aby używać kilku niezależnych serwerów Zabbix równolegle. Dozwolone są spacje.

Przykład dla proxy Zabbix:

```
ServerActive=127.0.0.1:10051
```

Przykład dla grupy proxy Zabbix:

```
ServerActive=proxy1.example.com;proxy2.example.com;proxy3.example.com;proxy4.example.com;proxy5.example.co
```

Przykład dla wielu serwerów:

```
ServerActive=127.0.0.1:20051,zabbix.domain,\[::1\]:30051,::1,\[12fc::1\]
```

Przykład dla wysokiej dostępności:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
```

Przykład dla wysokiej dostępności z dwoma klastrami i jednym serwerem:

```
ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051,zabbix.cluster2.node1;zabbix.cluster2.node2,z
```

SourceIP

Określ adres IP źródłowy dla:

- połączeń wychodzących do serwera Zabbix lub proxy Zabbix;
- nawiązywania połączeń podczas wykonywania niektórych pozycji (`web.page.get`, `net.tcp.port` itp.).

StatusPort

Ustaw port, na którym agent nasłuchuje żądań statusu HTTP (`http://localhost:<port>/status`).

Zakres: 1024-32767

Limit czasu

Określa, jak długo należy czekać (w sekundach) na nawiązanie połączenia i wymianę danych z proxy lub serwerem Zabbix.

Ten parametr definiuje czas trwania różnych operacji komunikacyjnych, w tym:

- oczekiwanie na odpowiedź z serwera Zabbix;
- wysyłanie żądań do serwera Zabbix, w tym żądań konfiguracji pozycji i danych pozycji w aktywnych kontrolach;
- pobieranie danych dziennika przez monitorowanie logfile lub dziennika zdarzeń systemu Windows;

- wysyłanie komunikatów heartbeat;
- maksymalny czas trwania kontroli `vfs.*`;
- pełnienie roli wartości awaryjnej w scenariuszach, w których serwer lub proxy w wersji starszej niż 7.0 wysyła kontrole bez limitów czasu.

Ten limit czasu **nie** będzie używany dla tych kontroli agenta, które mają konfigurowalne ustawienia limitu czasu we frontendzie (globalnie, na poziomie proxy lub dla pojedynczej pozycji).

Domyślnie: 3
 Zakres: 1-30

TLSAccept

Określ poziom szyfrowania wymagany dla połączeń przychodzących.
 Używane dla pasywnych kontroli.

Można podać wiele wartości, oddzielając je przecinkiem:

- *unencrypted* - akceptuj połączenia bez szyfrowania (domyślnie);
- *psk* - akceptuj połączenia z TLS i wstępnie współdzielonym kluczem (PSK);
- *cert* - akceptuj połączenia z TLS i certyfikatem.

Wymagane: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty najwyższego poziomu CA do weryfikacji certyfikatu peer, używaną do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Podaj pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używany do szyfrowanej komunikacji z komponentami Zabbix.

TLSCipherAll13

Określ ciąg szyfrów OpenSSL (TLS 1.3). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach i PSK.

Przykład:

```
TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSCipherCert13

Określ ciąg szyfrów OpenSSL (TLS 1.3). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na certyfikatach.

Należy pamiętać, że tego parametru nie można używać razem z `TLSAccept=cert,psk`; dla połączenia certyfikatowego (`TLSConnect=cert`) użyj zamiast tego `TLSCipherAll13`.

TLSCipherPSK13

Określ ciąg szyfrów OpenSSL (TLS 1.3). Zastąp domyślne kryteria wyboru zestawu szyfrów dla szyfrowania opartego na PSK.

Przykład:

```
TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
```

TLSConnect

Określ poziom szyfrowania wymagany dla połączeń agent z serwerem Zabbix lub proxy.
 Używane dla aktywnych kontroli.
 Można określić tylko jedną wartość: - *unencrypted* - połączenie bez szyfrowania (domyślnie); - *psk* - połączenie z użyciem TLS i wstępnie współdzielonego klucza (PSK); - *cert* - połączenie z użyciem TLS i certyfikatu.

Obowiązkowe: tak, jeśli zdefiniowano parametry certyfikatu TLS lub PSK (nawet dla połączenia *unencrypted*); w przeciwnym razie nie

TLSCRLFile

Określ pełną ścieżkę do pliku zawierającego unieważnione certyfikaty. Ten parametr jest używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSKeyFile

Podaj pełną ścieżkę do pliku zawierającego prywatny klucz agent, używany do szyfrowanej komunikacji między komponentami Zabbix.

TLSPSKFile

Określ pełną ścieżkę do pliku zawierającego współdzielony klucz agenta, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSPSKIdentity

Określ ciąg tożsamości pre-shared key, używany do szyfrowanej komunikacji z serwerem Zabbix lub proxy.

TLSServerCertIssuer

Określ dozwolonego wystawcę certyfikatu serwera (proxy).

TLSServerCertSubject

Określ dozwolony temat certyfikatu serwera (proxy).

UnsafeUserParameters

Zezwalaj na przekazywanie wszystkich znaków w argumentach do parametrów zdefiniowanych przez użytkownika. Następujące znaki są niedozwolone: \ ' " ` * ? [ ] { } ~ $! & ; ( ) < > | # @ %
Dodatkowo znaki nowej linii są niedozwolone.

Domyślnie: 0
 Wartości: 0 - nie zezwalaj, 1 - zezwalaj

UserParameter

Określ zdefiniowany przez użytkownika parametr do monitorowania. Może istnieć kilka takich parametrów.
Format: `UserParameter=<key>,<shell command>`
Pamiętaj, że polecenie powłoki nie może zwracać pustego ciągu ani wyłącznie znaku końca linii. Polecenia powłoki mogą używać ścieżek względnych, jeśli określono parametr `UserParameterDir`.

Przykład:

```
UserParameter=system.test,dir /b | find /c /v ""
UserParameter=check_cpu,.\custom\_script.bat
```

UserParameterDir

Określa domyślną ścieżkę wyszukiwania dla poleceń `UserParameter`. Jeśli jest używana, agent zmieni swój katalog roboczy na katalog określony tutaj przed wykonaniem polecenia. Dzięki temu polecenia UserParameter mogą używać względnego prefiksu `.\` zamiast pełnej ścieżki.
Dozwolony jest tylko jeden wpis.

Przykład:

```
UserParameterDir=C:\opt\myscripts
```

Parametry konfiguracji: Wtyczki

Zobacz parametry konfiguracji dla wtyczek Zabbix agent 2:

- wtyczka Ceph
- wtyczka Docker
- wtyczka Ember+
- wtyczka Memcached
- wtyczka Modbus
- wtyczka MongoDB
- wtyczka MQTT
- wtyczka MSSQL
- wtyczka MySQL
- wtyczka NVIDIA GPU
- wtyczka Oracle
- wtyczka PostgreSQL
- wtyczka Redis
- wtyczka S.M.A.R.T.

Zasady konfiguracji wtyczek

Ta strona opisuje zasady konfiguracji wtyczek i najlepsze praktyki.

Wszystkie wtyczki konfiguruje się za pomocą parametru `Plugins.*`, który może być częścią pliku konfiguracyjnego agenta 2 Zabbix lub pliku konfiguracyjnego wtyczki. Jeśli wtyczka korzysta z oddzielnego pliku konfiguracyjnego, ścieżkę do tego pliku należy określić w parametrze `Include` pliku konfiguracyjnego agenta 2 Zabbix.

Typowy parametr wtyczki ma następującą strukturę:

`Plugins.<PluginName>.<Parameter>=<Value>`

Ponadto istnieją dwie określone grupy parametrów:

- `Plugins.<PluginName>.Default.<Parameter>=<Value>` służy do definiowania domyślnych wartości parametrów.
- `Plugins.<PluginName>.<SessionName>.<Parameter>=<Value>` służy do definiowania oddzielnych zestawów parametrów dla różnych celów monitorowania za pomocą nazwanych sesji.

Wszystkie nazwy parametrów powinny spełniać następujące wymagania:

- Zaleca się rozpoczynanie nazw wtyczek wielką literą.
- Parametr powinien rozpoczynać się wielką literą.
- Znaki specjalne są niedozwolone.
- Zagnieżdżanie nie jest ograniczone maksymalnym poziomem.
- Liczba parametrów nie jest ograniczona.

Na przykład, aby wykonywać aktywne sprawdzenia, które nie mają ustawionego *Scheduling* interwału aktualizacji bezpośrednio po ponownym uruchomieniu agenta, tylko dla wtyczki Uptime, ustaw `Plugins.Uptime.System.ForceActiveChecksOnStart=1` w pliku konfiguracyjnym. Podobnie, aby ustawić niestandardowy limit dla współbieżnych sprawdzeń wtyczki CPU, ustaw `Plugins.CPU.System.Capacity=N` w pliku konfiguracyjnym.

Wartości domyślne

Możesz ustawić wartości domyślne dla parametrów związanych z połączeniem (URI, nazwa użytkownika, hasło itp.) w pliku konfiguracyjnym w formacie:

```
Plugins.<PluginName>.Default.<Parameter>=<Value>
```

Na przykład: `Plugins.Mysql.Default.Username=zabbix`, `Plugins.MongoDB.Default.Uri=tcp://127.0.0.1:27017` itd.

Jeśli wartość takiego parametru nie zostanie podana w kluczu pozycji ani w parametrach nazwanej sesji, wtyczka użyje wartości domyślnej. Jeśli parametr domyślny również nie jest zdefiniowany, zostaną użyte wartości domyślne zapisane w kodzie.

> **Note:**
> Jeśli klucz pozycji nie zawiera żadnych parametrów, agent 2 Zabbix spróbuje zebrać metrykę przy użyciu wartości zdefiniowanych w sekcji parametrów domyślnych.

Nazwane sesje

Nazwane sesje stanowią dodatkowy poziom parametrów wtyczki i mogą być używane do określania oddzielnych zestawów parametrów uwierzytelniania dla każdej z monitorowanych instancji. Każdy parametr nazwanej sesji powinien mieć następującą strukturę:

```
Plugins.<PluginName>.Sessions.<SessionName>.<Parameter>=<Value>
```

Nazwa sesji może być użyta jako kluczowy parametr elementu connString zamiast osobnego podawania URI, nazwy użytkownika i/lub hasła.

W kluczach pozycji pierwszy parametr może być albo connString, albo URI. Jeśli pierwszy parametr klucza nie pasuje do żadnej nazwy sesji, zostanie potraktowany jako URI. Należy pamiętać, że przekazywanie osadzonych poświadczeń URI w kluczu pozycji nie jest obsługiwane; zamiast tego należy używać parametrów nazwanej sesji.

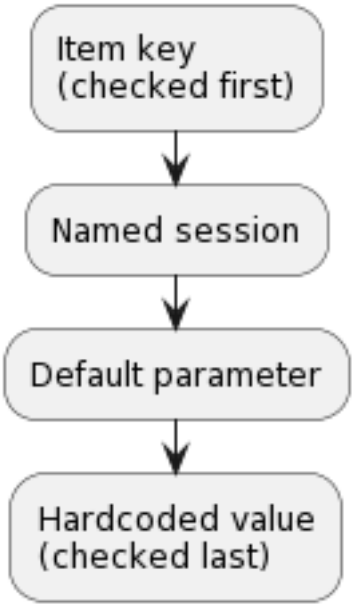
Lista dostępnych parametrów nazwanej sesji zależy od wtyczki.

Możliwe jest zastąpienie parametrów sesji przez określenie nowych wartości w parametrach klucza pozycji (zobacz przykład).

Jeśli parametr nie jest zdefiniowany dla nazwanej sesji, Zabbix agent 2 użyje wartości zdefiniowanej w domyślnym parametrze wtyczki.

Priorytet parametrów

Wtyczki Zabbix agent 2 wyszukują wartości parametrów związanych z połączeniem w następującej kolejności:



1. Pierwszy parametr klucza pozycji jest porównywany z nazwami sesji. Jeśli nie zostanie znalezione dopasowanie, jest on traktowany jako rzeczywista wartość; w takim przypadku krok 3 zostanie pominięty. Jeśli zostanie znalezione dopasowanie, wartość parametru (zwykle URI) musi być zdefiniowana w nazwanej sesji.
2. Pozostałe parametry zostaną pobrane z klucza pozycji, jeśli są zdefiniowane.
3. Jeśli parametr klucza pozycji (na przykład password) jest pusty, wtyczka wyszuka odpowiadający mu parametr nazwanej sesji.
4. Jeśli parametr sesji również nie jest określony, zostanie użyta wartość zdefiniowana w odpowiednim domyślnym parametrze.
5. Jeśli wszystko inne zawiedzie, wtyczka użyje zakodowanej na stałe wartości domyślnej.

Przykład 1

Monitorowanie dwóch instancji "MySQL1" i "MySQL2".

Parametry konfiguracji:

```
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
```

W wyniku tej konfiguracji każda nazwa sesji może być używana jako connString w kluczu pozycji, np. `mysql.ping[MySQL1]` lub `mysql.ping[MySQL2]`.

Przykład 2

Podanie niektórych parametrów w kluczu pozycji.

Parametry konfiguracji:

```
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
```

Klucz pozycji: `pgsql.ping[session1,new_username,,postgres]`

W wyniku tej konfiguracji agent połączy się z PostgreSQL, używając następujących parametrów:

- URI z parametru sesji: `192.0.2.234:5432`
- Nazwa użytkownika z klucza pozycji: `new_username`
- Hasło z parametru sesji (ponieważ zostało pominięte w kluczu pozycji): `session_password`
- Nazwa bazy danych z klucza pozycji: `postgres`

Przykład 3

Zbieranie metryki przy użyciu domyślnych parametrów konfiguracji.

Parametry konfiguracji:

```
Plugins.PostgreSQL.Default.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Default.User=zabbix
Plugins.PostgreSQL.Default.Password=password
```

Klucz pozycji: `pgsql.ping[,,,postgres]`

W wyniku tej konfiguracji agent połączy się z PostgreSQL, używając następujących parametrów:

- Domyślny URI: `192.0.2.234:5432`
- Domyślna nazwa użytkownika: `zabbix`
- Domyślne hasło: `password`
- Nazwa bazy danych z klucza pozycji: `postgres`

Połączenia

Niektóre wtyczki obsługują jednoczesne zbieranie metryk z wielu instancji. Można monitorować zarówno instancje lokalne, jak i zdalne. Obsługiwane są połączenia TCP oraz Unix-socket.

Zaleca się skonfigurowanie wtyczek tak, aby utrzymywały połączenia z instancjami w stanie otwartym. Zaletami są mniejsze przeciążenie sieci, niższe opóźnienia oraz mniejsze użycie CPU i pamięci dzięki mniejszej liczbie połączeń. Odpowiada za to biblioteka klienta.

> **Note:**
> Okres, przez jaki nieużywane połączenia powinny pozostawać otwarte, można określić za pomocą parametru `Plugins.<PluginName>.KeepAlive`. Przykład: `Plugins.Memcached.KeepAlive`

Wtyczka Ceph

Przegląd

Parametry konfiguracyjne umożliwiają dostosowanie zachowania wtyczki Ceph.

Zaleca się określanie ich we własnym pliku konfiguracyjnym (np. `ceph.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Ceph.Default.ApiKey	Określa domyślny klucz API do łączenia z Ceph.
Plugins.Ceph.Default.Mode	Określa domyślny tryb łączenia z Ceph.
Plugins.Ceph.Default.URI	Określa domyślny URI do łączenia z Ceph.
Plugins.Ceph.Default.User	Określa domyślną nazwę użytkownika do łączenia z Ceph.
Plugins.Ceph.InsecureSkipVerify	Określa, czy klient HTTP powinien weryfikować łańcuch certyfikatów serwera i nazwę hosta.
Plugins.Ceph.KeepAlive	Ustawia maksymalny czas oczekiwania przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Ceph.Sessions.<Session name>.ApiKey	Określa klucz API dla nazwanej sesji do łączenia z Ceph.
Plugins.Ceph.Sessions.<Session name>.Mode	Określa tryb nazwanej sesji do łączenia z Ceph.
Plugins.Ceph.Sessions.<Session name>.URI	Określa URI nazwanej sesji do łączenia z Ceph.
Plugins.Ceph.Sessions.<Session name>.User	Określa nazwę użytkownika nazwanej sesji do łączenia z Ceph.
Plugins.Ceph.Timeout	Ustawia limit czasu wykonania żądania.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Ceph.Default.ApiKey

Określ domyślny klucz API do łączenia z Ceph.
 Używany, jeśli nie określono żadnej wartości w kluczu pozycji lub nazwanej sesji.

Plugins.Ceph.Default.Mode

Określ domyślny tryb łączenia z Ceph.
 Tryb natywny jest obsługiwany tylko w systemie Linux i począwszy od Ceph 16. Tryb restful nie będzie działać z Ceph w wersji 20 lub nowszej.
 Należy pamiętać, że zestaw poświadczeń użytkownika różni się dla każdego trybu i tryby te nie są ze sobą zgodne.

Domyślnie: `restful` (przestarzałe)
 Zakres: native/restful

Plugins.Ceph.Default.URI

Określa domyślny URI do łączenia się z Ceph.
 Używany, jeśli nie podano wartości w kluczu pozycja lub nazwanej sesji.

Nie powinien zawierać osadzonych poświadczeń (zostaną zignorowane).
Musi odpowiadać formatowi URI.
Obsługiwany jest tylko schemat `https`; schemat można pominąć.
Port można pominąć (domyślnie=8003).

Domyślnie: `https://localhost:8003`

Przykłady:

```
https://127.0.0.1:8003
localhost
```

Plugins.Ceph.Default.User

Określ domyślną nazwę użytkownika do łączenia z Ceph.
 Używane, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.

Plugins.Ceph.InsecureSkipVerify

Określa, czy klient HTTP powinien weryfikować łańcuch certyfikatów serwera oraz nazwę hosta.
 Jeśli *true*, TLS akceptuje dowolny certyfikat przedstawiony przez serwer oraz dowolną nazwę hosta w tym certyfikacie. W tym trybie TLS jest podatny na ataki typu man-in-the-middle (używaj tylko do testów).

Domyślnie: `false`
 Zakres: false/true

Plugins.Ceph.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60-900

Plugins.Ceph.Sessions.<SessionName>.ApiKey

Określ nazwany klucz API sesji do łączenia z Ceph.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Ceph.Sessions.<SessionName>.Mode

Określa nazwany tryb sesji do łączenia z Ceph.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Tryb native jest obsługiwany tylko w systemie Linux oraz począwszy od Ceph 16. Tryb restful nie będzie działać z wersją Ceph 20 lub nowszą.
 Należy pamiętać, że zestaw poświadczeń użytkownika różni się dla każdego trybu i tryby te nie są ze sobą zgodne.

Domyślnie: `restful` (przestarzałe)
 Zakres: native/restful

Plugins.Ceph.Sessions.<SessionName>.URI

Określ URI nazwanej sesji do łączenia z Ceph.
 <SessionName> - określ nazwę sesji do użycia w kluczach pozycji.
Nie powinno zawierać osadzonych poświadczeń (zostaną zignorowane).
Musi odpowiadać formatowi URI.
 Obsługiwany jest tylko schemat `https`; schemat można pominąć.
Port można pominąć (domyślnie=8003).

Domyślnie: `https://localhost:8003`

Przykłady:

```
https://127.0.0.1:8003
localhost
```

Plugins.Ceph.Sessions.<SessionName>.User

Określ nazwę użytkownika dla wskazanej sesji używanej do łączenia z Ceph.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Ceph.Timeout

Ustaw limit czasu wykonania żądania (czas oczekiwania na zakończenie żądania przed jego przerwaniem) w sekundach.

Domyślnie: `global timeout<br>` Zakres: 1-30

Zobacz także

- Instrukcje dotyczące konfigurowania wtyczek
- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)

Wtyczka Docker

Przegląd

Parametry konfiguracji umożliwiają dostosowanie zachowania wtyczki Docker.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `docker.conf`), a nie w pliku konfiguracji agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Docker.Endpoint	Określa lokalizację gniazda UNIX demona Docker..
Plugins.Docker.Timeout	Określa limit czasu wykonania żądania.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Docker.Endpoint

Określ lokalizację gniazda UNIX demona Docker.
 Musi zawierać schemat (obsługiwany jest tylko `unix://`).

Domyślnie: `unix:///var/run/docker.sock`

Plugins.Docker.Timeout

Określ limit czasu wykonania żądania (czas w sekundach, przez jaki należy czekać na zakończenie żądania przed jego przerwaniem).

Domyślnie: `global timeout<br>` Zakres: 1–30

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje dotyczące konfiguracji wtyczek

Wtyczka Ember+

Przegląd

Parametry konfiguracyjne umożliwiają dostosowanie działania wtyczki Ember+.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `emberplus.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie należy użyć dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wtyczka Ember+ jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki Ember+.

Ta wtyczka jest obecnie dostępna wyłącznie do kompilacji ze źródeł (zarówno dla Unix, jak i Windows).

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.EmberPlus.Default.Uri	Określa domyślny URI do łączenia z Ember+.
Plugins.EmberPlus.KeepAlive	Ustawia maksymalny czas oczekiwania przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.EmberPlus.Sessions.<SessionName>.Uri	Określa URI nazwanej sesji do łączenia z Ember+.
Plugins.EmberPlus.System.Path	Określa ścieżkę do pliku wykonywalnego wtyczki Ember+.
Plugins.EmberPlus.Timeout	Ustawia limit czasu wykonania żądania.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.EmberPlus.Default.Uri

Określ domyślny URI do łączenia z Ember+.
 Jedynym obsługiwanym schematem jest `tcp://`. Schemat można pominąć. Osadzone poświadczenia zostaną zignorowane.

Domyślnie: `tcp://localhost:9998`

Plugins.EmberPlus.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60–900

Plugins.EmberPlus.Sessions.<SessionName>.Uri

Określ URI nazwanej sesji do połączenia z Ember+.
 Jedynym obsługiwanym schematem jest `tcp://`. Schemat można pominąć. Osadzone poświadczenia zostaną zignorowane.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Domyślnie: `tcp://localhost:9998`

Plugins.EmberPlus.System.Path

Określ ścieżkę do pliku wykonywalnego wtyczki Ember+.

Przykład: `Plugins.EmberPlus.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-ember-plus`

Plugins.EmberPlus.Timeout

Ustaw czas oczekiwania, w sekundach, na odpowiedź serwera podczas pierwszego połączenia oraz podczas kolejnych operacji w sesji.

Domyślnie: `global timeout`
 Zakres: 1–30

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki i informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).
-t, --test <item key>	Uruchamia wtyczkę do testowania (konfiguracja wtyczki jest ignorowana).

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka Memcached

Przegląd

Parametry konfiguracyjne umożliwiają dostosowanie działania wtyczki Memcached.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `memcached.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Memcached.Default.Password	Określa domyślne hasło do łączenia z Memcached.
Plugins.Memcached.Default.Uri	Określa domyślny URI do łączenia z Memcached.
Plugins.Memcached.Default.User	Określa domyślną nazwę użytkownika do łączenia z Memcached.
Plugins.Memcached.KeepAlive	Ustawia maksymalny czas oczekiwania przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Memcached.Sessions.<SessionName>.Password	Określa hasło dla nazwanej sesji do łączenia z Memcached.
Plugins.Memcached.Sessions.<SessionName>.Uri	Określa URI dla nazwanej sesji do łączenia z Memcached.
Plugins.Memcached.Sessions.<SessionName>.User	Określa nazwę użytkownika dla nazwanej sesji do łączenia z Memcached.
Plugins.Memcached.Timeout	Ustawia limit czasu wykonania żądania.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Memcached.Default.Password

Określ domyślne hasło do łączenia z Memcached.
 Używane, jeśli nie podano wartości w kluczu pozycji lub nazwanej sesji.

Plugins.Memcached.Default.Uri

Określa domyślny URI do połączenia z Memcached.
 Używany, jeśli nie określono wartości w kluczu pozycja lub nazwanej sesji.
 Osadzone poświadczenia zostaną zignorowane.
 Musi odpowiadać formatowi URI. Obsługiwane schematy: `tcp`, `unix`.
 Schemat można pominąć.
 Port można pominąć (domyślnie=11211).

Domyślnie: `tcp://localhost:11211`

Przykłady:

```
tcp://localhost:11211
localhost
unix:/var/run/memcached.sock
```

Plugins.Memcached.Default.User

Określ domyślną nazwę użytkownika do łączenia z Memcached.
 Używana, jeśli w kluczu pozycji lub nazwanej sesji nie zostanie określona żadna wartość.

Plugins.Memcached.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60-900

Plugins.Memcached.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji używane do łączenia z Memcached.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.

Plugins.Memcached.Sessions.<SessionName>.Uri

Określa URI nazwanej sesji do łączenia z Memcached.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.
 Osadzone poświadczenia zostaną zignorowane.
 Musi odpowiadać formatowi URI. Obsługiwane schematy: `tcp`, `unix`.
 Schemat można pominąć.
 Port można pominąć (domyślnie=11211).

Przykłady:

```
tcp://localhost:11211
localhost
unix:/var/run/memcached.sock
```

Plugins.Memcached.Sessions.<SessionName>.User

Określ nazwę użytkownika dla wskazanej sesji używanej do łączenia z Memcached.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.

Plugins.Memcached.Timeout

Ustaw limit czasu wykonania żądania (czas w sekundach, przez który należy czekać na zakończenie żądania przed jego przerwaniem).

Domyślnie: `global timeout`
 Zakres: 1–30

Zobacz także:

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka Modbus

Przegląd

Parametry konfiguracyjne umożliwiają dostosowanie zachowania wtyczki Modbus.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `modbus.conf`), a nie w pliku konfiguracyjnym agent 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Modbus.Sessions.<SessionName>.Endpoint	Określa punkt końcowy nazwanej sesji do łączenia z Modbus.
Plugins.Modbus.Sessions.<SessionName>.SlaveID	Określa identyfikator Slave ID nazwanej sesji do łączenia z Modbus.
Plugins.Modbus.Sessions.<SessionName>.Timeout	Ustawia limit czasu dla nazwanej sesji w sekundach.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Modbus.Sessions.<SessionName>.Endpoint

Określa nazwany punkt końcowy sesji do łączenia z Modbus.
 Ciąg połączenia składa się ze schematu protokołu, adresu hosta oraz portu lub nazwy portu szeregowego i atrybutów.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Modbus.Sessions.<SessionName>.SlaveID

Określ identyfikator slave dla nazwanej sesji używanej do połączenia z Modbus.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Przykład: `Plugins.Modbus.Sessions.MB1.SlaveID=20`

Uwaga ten parametr nazwanej sesji jest sprawdzany tylko wtedy, gdy wartość podana w parametrze slave ID klucza pozycji jest pusta.

Plugins.Modbus.Sessions.<SessionName>.Timeout

Ustaw nazwany limit czasu sesji w sekundach.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Przykład: `Plugins.Modbus.Sessions.MB1.Timeout=2`

Jeśli musisz ustawić limit czasu wykonania żądania (jak długo czekać na zakończenie żądania przed jego przerwaniem), użyj formularza konfiguracji pozycji.

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka MongoDB

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania wtyczki MongoDB.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `mongodb.conf`), a nie w pliku konfiguracji agenta 2. Następnie należy użyć dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wtyczka MongoDB jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki MongoDB.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Podczas łączenia z MongoDB wtyczka sprawdza swoją konfigurację w określonej kolejności, która decyduje o tym, której wartości użyć. Kolejność, od najwyższego do najniższego priorytetu, jest następująca:

1. Parametry określone w URI połączenia (np. parametry zapytania zdefiniowane po ? w `Plugins.MongoDB.Sessions.<session_name`
mają najwyższy priorytet).
2. Parametry w kluczu pozycji Zabbix agent 2 (np. mongodb.collection.stats).
3. Parametry zdefiniowane w nazwanej sesji (np. `Plugins.MongoDB.Sessions.<SessionName>.User`, `Plugins.MongoDB.Session`
4. Wartości domyślne (np. `Plugins.MongoDB.Default.User`, `Plugins.MongoDB.Default.Password`).

Na przykład:

- Jeśli określisz bazę danych do uwierzytelniania połączenia z serwerem MongoDB w parametrze `*.<SessionName>.Uri` (np. `Plugins.MongoDB.Sessions.<session_name>.Uri=mongodb://user:password@127.0.0.1:27017/?authSource=admin`
wtyczka uzna, że parametr `*.<SessionName>.AuthSource` ma wartość `admin`, nawet jeśli parametr `*.<SessionName>.AuthSour`
jest ustawiony na inną wartość.
- Jeśli określisz URI w parametrze `*.<SessionName>.Uri` (np. `Plugins.MongoDB.Sessions.<session_name>.Uri=mongodb://1`
ale podasz nazwę użytkownika i hasło w kluczu pozycji Zabbix agent 2, wtyczka użyje nazwy użytkownika i hasła z klucza pozycji, nawet jeśli skonfigurowano `Plugins.MongoDB.Sessions.<SessionName>.User` i `Plugins.MongoDB.Sessions.<SessionN`

Attention:
Jeśli używany jest schemat `mongodb+srv://`, URI może pobierać parametry z rekordu DNS TXT. Parametry uzyskane z tego rekordu TXT zastępują wszystkie inne konfiguracje, w tym parametry URI, parametry klucza pozycji Zabbix Agent 2 oraz parametry sesji.
 Dla nazwy hosta użytej w ciągu połączenia może istnieć tylko jeden rekord TXT.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.MongoDB.Default.Auth	Określa domyślny mechanizm uwierzytelniania do łączenia z serwerem MongoDB.
Plugins.MongoDB.Default.AuthSource	Określa domyślną bazę danych do uwierzytelniania połączenia z serwerem MongoDB.
Plugins.MongoDB.Default.Disable	Wyłącza automatyczne wykrywanie węzłów zestawu replik MongoDB.
Plugins.MongoDB.Default.Password	Określa domyślne hasło.
Plugins.MongoDB.Default.ReplicaSet	Określa domyślną nazwę zestawu replik, używaną do automatycznego wykrywania węzłów zestawu replik MongoDB.
Plugins.MongoDB.Default.ReadPreference	Określa domyślną kolejność łączenia z węzłami zestawu replik MongoDB lub tryb awaryjny.
Plugins.MongoDB.Default.TLSConnect	Określa poziom szyfrowania wymagany dla komunikacji między Zabbix agent 2 a monitorowanymi bazami danych.
Plugins.MongoDB.Default.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer.
Plugins.MongoDB.Default.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów.
Plugins.MongoDB.Default.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych.
Plugins.MongoDB.Default.Uri	Określa domyślne URI.
Plugins.MongoDB.Default.User	Określa domyślną nazwę użytkownika.
Plugins.MongoDB.KeepAlive	Ustawia maksymalny czas oczekiwania przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.MongoDB.Sessions.<SessionName>.Auth	Określa mechanizm uwierzytelniania nazwanej sesji do łączenia z serwerem MongoDB.
Plugins.MongoDB.Sessions.<SessionName>.AuthSource	Określa bazę danych nazwanej sesji do uwierzytelniania połączenia z serwerem MongoDB.
Plugins.MongoDB.Sessions.<SessionName>.Disable	Wyłącza automatyczne wykrywanie węzłów zestawu replik MongoDB dla nazwanej sesji.
Plugins.MongoDB.Sessions.<SessionName>.Password	Określa hasło nazwanej sesji.

Parameter	Description
Plugins.MongoDB.Sessions.<SessionName>.ReplicaSet	Określa nazwę zestawu replik nazwanej sesji, używaną do automatycznego wykrywania węzłów zestawu replik MongoDB.
Plugins.MongoDB.Sessions.<SessionName>.ReadPreference	Określa kolejność nawiązywania sesji dla łączenia z węzłami zestawu replik MongoDB lub tryb awaryjny.
Plugins.MongoDB.Sessions.<SessionName>.TLSConnect	Określa poziom szyfrowania wymagany dla komunikacji między Zabbix agent 2 a monitorowanymi bazami danych dla nazwanej sesji.
Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer dla nazwanej sesji.
Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów dla nazwanej sesji.
Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych dla nazwanej sesji.
Plugins.MongoDB.Sessions.<SessionName>.Uri	Określa URI nazwanej sesji.
Plugins.MongoDB.Sessions.<SessionName>.User	Określa nazwę użytkownika nazwanej sesji.
Plugins.MongoDB.System.Path	Określa ścieżkę do pliku wykonywalnego wtyczki.
Plugins.MongoDB.Timeout	Ustawia limit czasu wykonania żądania.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze rozpoczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.MongoDB.Default.AuthMechanism

Określ domyślny mechanizm uwierzytelniania do łączenia się z serwerem MongoDB.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres: `SCRAM-SHA-1`, `SCRAM-SHA-256` lub `MONGODB-X509`

Plugins.MongoDB.Default.AuthSource

Określ domyślną bazę danych do uwierzytelniania połączenia z serwerem MongoDB.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.

Domyślnie: `admin`

Jeśli używany jest mechanizm uwierzytelniania `MONGODB-X509`, ten parametr jest automatycznie ustawiany na `$external`; zmiana go na jakąkolwiek inną wartość spowoduje błąd podczas uruchamiania.

Plugins.MongoDB.Default.Direct

Wyłącz automatyczne wykrywanie węzłów zestawu replik MongoDB.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Domyślnie: `false`
 Zakres: `true` - łączy się tylko z określonym serwerem, ignorując innych członków zestawu replik, `false` - automatycznie wykrywa wszystkie węzły w zestawie replik.

Jeśli używany jest schemat `tcp://` (przestarzały), parametr jest domyślnie ustawiany na `true`.

Plugins.MongoDB.Default.Password

Określ domyślne hasło.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Plugins.MongoDB.Default.ReplicaSet

Określ domyślną nazwę zestawu replik, używaną do automatycznego wykrywania węzłów zestawu replik MongoDB.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Jeśli ten parametr jest używany ze schematem `mongodb+srv://`, będzie działał tak, jakby podano wiele adresów URL; serwer DNS może mieć rekord TXT, który może zastąpić ten (lub dowolny inny) parametr.
 Niedostępne, jeśli parametr `Plugins.MongoDB.Sessions.<SessionName>.Direct` jest ustawiony na `true`; agent nie uruchomi się.

Plugins.MongoDB.Default.ReadPreference

Określ domyślną kolejność nawiązywania połączeń z węzłami zestawu replik MongoDB lub tryb awaryjny.
 <SessionName> - określ nazwę sesji do użycia w kluczach pozycji.
 Niedostępne, jeśli parametr `Plugins.MongoDB.Sessions.<SessionName>.Direct` jest ustawiony na `true`; agent nie uruchomi się.

Zakres: primary, primaryPreferred, secondary, secondaryPreferred, nearest, PrimaryMode, PrimaryPreferredMode, SecondaryMode, SecondaryPreferredMode lub NearestMode

Zalecane są wartości primary, primaryPreferred, secondary, secondaryPreferred i nearest. Wartości kończące się na Mode są obsługiwane jako aliasy zgodności ze starszymi wersjami.

Plugins.MongoDB.Default.TLSConnect

Określa poziom szyfrowania wymagany dla komunikacji między agent 2 Zabbix a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Obsługiwane od wersji wtyczki 1.2.1.

Zakres: required - wymagaj połączenia TLS, verify_ca - weryfikuj certyfikaty, verify_full - weryfikuj certyfikaty i adres IP

Plugins.MongoDB.Default.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównego urzędu CA używane do weryfikacji certyfikatu peer, wykorzystywane w szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Wymagane: yes, jeśli Plugins.MongoDB.Sessions.<SessionName>.TLSConnect jest ustawione na verify_ca lub verify_full

Plugins.MongoDB.Default.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używany do szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Obowiązkowe: yes, jeśli określono Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile

Plugins.MongoDB.Default.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych, używany do szyfrowanej komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Obowiązkowe: yes, jeśli określono Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile

Plugins.MongoDB.Default.Uri

Określ domyślny URI.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
 Musi odpowiadać formatowi URI.
 Obsługiwane schematy: mongodb:// i mongodb+srv://; schemat tcp:// jest przestarzały i zachowany ze względu na zgodność wsteczną z istniejącymi konfiguracjami.
 Port można pominąć (domyślnie=27017).
Przykłady: “

Domyślnie: tcp://127.0.0.1:27017

Przykłady:

```
mongodb://127.0.0.1:27017
mongodb+srv://example.com
localhost
```

Plugins.MongoDB.Default.User

Określ domyślną nazwę użytkownika.
 <SessionName> - określ nazwę sesji do użycia w kluczach pozycji.

Plugins.MongoDB.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60-900

Plugins.MongoDB.Sessions.<SessionName>.AuthMechanism

Określ nazwany mechanizm uwierzytelniania sesji do łączenia z serwerem MongoDB.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres: SCRAM-SHA-1, SCRAM-SHA-256 lub MONGODB-X509

Plugins.MongoDB.Sessions.<SessionName>.AuthSource

Określa nazwaną bazę danych sesji do uwierzytelniania połączenia z serwerem MongoDB.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Domyślnie: admin

Jeśli używany jest mechanizm uwierzytelniania MONGODB-X509, ten parametr jest automatycznie ustawiany na $external; zmiana go na jakąkolwiek inną wartość spowoduje błąd podczas uruchamiania.

Plugins.MongoDB.Sessions.<SessionName>.Direct

Wyłącz automatyczne wykrywanie węzłów repliki MongoDB dla nazwanego sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Domyślnie: `false`
 Zakres: `true` - łączy się tylko z określonym serwerem, ignorując innych członków zestawu replik, `false` - automatycznie wykrywa wszystkie węzły w zestawie replik.

Jeśli używany jest schemat `tcp://` (przestarzały), parametr jest domyślnie ustawiony na `true`.

Plugins.MongoDB.Sessions.<SessionName>.Password

Określa hasło dla nazwanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MongoDB.Sessions.<SessionName>.ReplicaSet

Określa nazwę zestawu replik dla nazwanej sesji, używaną do automatycznego wykrywania węzłów zestawu replik MongoDB.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Jeśli ten parametr jest używany ze schematem `mongodb+srv://`, będzie działał tak, jakby podano wiele adresów URL; serwer DNS może mieć rekord TXT, który może zastąpić ten (lub dowolny inny) parametr.
 Niedostępne, jeśli parametr `Plugins.MongoDB.Sessions.<SessionName>.Direct` jest ustawiony na `true`; agent nie uruchomi się.

Plugins.MongoDB.Sessions.<SessionName>.ReadPreference

Określ kolejność nazwanej sesji podczas łączenia z węzłami zestawu replik MongoDB lub używania trybu awaryjnego.
 <SessionName> - określ nazwę sesji do użycia w kluczach pozycji.
 Niedostępne, jeśli parametr `Plugins.MongoDB.Sessions.<SessionNa` jest ustawiony na `true`; agent nie uruchomi się.

Zakres: `primary`, `primaryPreferred`, `secondary`, `secondaryPreferred`, `nearest`, `PrimaryMode`, `PrimaryPreferredMode`, `SecondaryMode`, `SecondaryPreferredMode` lub `NearestMode`

Zalecane są wartości `primary`, `primaryPreferred`, `secondary`, `secondaryPreferred` i `nearest`. Wartości kończące się na `Mode` są obsługiwane jako aliasy zapewniające zgodność ze starszymi wersjami.

Plugins.MongoDB.Sessions.<SessionName>.TLSConnect

Określ poziom szyfrowania wymagany dla komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych dla wskazanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Obsługiwane od wersji wtyczki 1.2.1.

Zakres: `required` - wymagaj połączenia TLS, `verify_ca` - weryfikuj certyfikaty, `verify_full` - weryfikuj certyfikaty i adres IP

Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używane w szyfrowanej komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych dla nazwanej sesji.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Wymagane: `yes`, jeśli `Plugins.MongoDB.Sessions.<SessionName>.TLSConnect` jest ustawione na `verify_ca` lub `verify_full`

Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używany do szyfrowanej komunikacji między agent 2 Zabbix a monitorowanymi bazami danych dla nazwanej sesji.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Obowiązkowe: `yes`, jeśli określono `Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile`

Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych, używany do szyfrowanej komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych dla nazwanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Obowiązkowe: `yes`, jeśli określono `Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile`

Plugins.MongoDB.Sessions.<SessionName>.Uri

Określa URI nazwanej sesji.
 <SessionName> - definiuje nazwę sesji do użycia w kluczach pozycji.
 Musi odpowiadać formatowi URI.
 Obsługiwane schematy: `mongodb://` i `mongodb+srv://`; schemat `tcp://` jest przestarzały i zachowany ze względu na zgodność wsteczną z istniejącymi konfiguracjami.
 Port można pominąć (domyślnie=27017).
Przykłady: “

Domyślnie: `tcp://127.0.0.1:27017`

Przykłady:

```
mongodb://127.0.0.1:27017
mongodb+srv://example.com
localhost
```

Plugins.MongoDB.Sessions.<SessionName>.User

Określa nazwę użytkownika dla wskazanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MongoDB.System.Path

Określ ścieżkę do pliku wykonywalnego wtyczki MongoDB.

Przykład użycia: `Plugins.MongoDB.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-mongodb`

Plugins.MongoDB.Timeout

Ustaw limit czasu wykonania żądania (czas w sekundach, przez który należy czekać na zakończenie żądania przed jego przerwaniem).

Domyślnie: `global timeout`
 Zakres: 1-30

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki i informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).
-t, --test <item key>	Uruchamia wtyczkę do testowania (konfiguracja wtyczki jest ignorowana).

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka MQTT

Omówienie

Parametry konfiguracyjne umożliwiają dostosowanie zachowania wtyczki MQTT.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `mqtt.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Omówienie parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.MQTT.Default.Password	Określa domyślne hasło do łączenia z MQTT.
Plugins.MQTT.Default.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer.
Plugins.MQTT.Default.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów.
Plugins.MQTT.Default.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz MQTT.
Plugins.MQTT.Default.Topic	Określa domyślny temat dla subskrypcji MQTT.
Plugins.MQTT.Default.Url	Określa domyślny ciąg połączenia z brokerem MQTT.
Plugins.MQTT.Default.User	Określa domyślną nazwę użytkownika do łączenia z MQTT.
Plugins.MQTT.Sessions.<SessionName>.Password	Określa hasło dla nazwanej sesji do łączenia z MQTT.
Plugins.MQTT.Sessions.<SessionName>.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer dla nazwanej sesji.
Plugins.MQTT.Sessions.<SessionName>.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów dla nazwanej sesji.
Plugins.MQTT.Sessions.<SessionName>.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz MQTT dla nazwanej sesji.
Plugins.MQTT.Sessions.<SessionName>.Topic	Określa temat nazwanej sesji dla subskrypcji MQTT.
Plugins.MQTT.Sessions.<SessionName>.Url	Określa ciąg połączenia z brokerem MQTT dla nazwanej sesji.

Parameter	Description
Plugins.MQTT.Sessions.<SessionName>.User	Określa nazwę użytkownika dla nazwanej sesji do łączenia z MQTT.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.MQTT.Default.Password

Określ domyślne hasło do połączenia z MQTT.
 Używane, jeśli nie podano wartości w kluczu pozycji lub nazwanej sesji.

Plugins.MQTT.Default.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer dla szyfrowanej komunikacji między agent 2 Zabbix a brokerem MQTT.
 Używane, jeśli w nazwanej sesji nie określono żadnej wartości.

Plugins.MQTT.Default.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów do szyfrowanej komunikacji między agentem Zabbix 2 a brokerem MQTT.
 Używane, jeśli w nazwanej sesji nie określono żadnej wartości.

Plugins.MQTT.Default.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz MQTT do szyfrowanej komunikacji między agentem Zabbix agent 2 a brokerem MQTT.
 Używane, jeśli w nazwanej sesji nie określono żadnej wartości.

Plugins.MQTT.Default.Topic

Określa domyślny temat dla subskrypcji MQTT.
 Używany, jeśli w kluczu pozycji lub nazwanej sesji nie zostanie określona żadna wartość.
 Temat może zawierać symbole wieloznaczne (”+”,”#”).

Przykłady:

```
path/to/file
path/to/#
path/+/topic
```

Plugins.MQTT.Default.Url

Określa domyślny ciąg połączenia z brokerem MQTT.
 Używany, jeśli w kluczu pozycja lub nazwanej sesji nie określono żadnej wartości.
 Nie powinien zawierać parametrów zapytania.
 Musi odpowiadać formatowi URL.
 Obsługiwane schematy: `tcp` (domyślny), `ws`, `tls`
 Schemat można pominąć.
 Port można pominąć (domyślnie=1883)

Domyślnie: `tcp://localhost:1883`

Przykłady:

```
tcp://host:1883
localhost
ws://host:8080
```

Plugins.MQTT.Default.User

Określa domyślną nazwę użytkownika do łączenia z MQTT.
 Używana, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.

Plugins.MQTT.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji używane do połączenia z MQTT.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MQTT.Sessions.<SessionName>.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer dla nazwanej sesji, używanej do szyfrowanej komunikacji między agentem Zabbix 2 a brokerem MQTT.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MQTT.Sessions.<SessionName>.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów dla nazwanego sesji, używanego do szyfrowanej komunikacji między Zabbix agent 2 a brokerem MQTT.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MQTT.Sessions.<SessionName>.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz MQTT dla nazwanego sesji, używanego do szyfrowanej komunikacji między agentem Zabbix 2 a brokerem MQTT.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MQTT.Sessions.<SessionName>.Topic

Określa nazwany temat sesji dla subskrypcji MQTT.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Temat może zawierać symbole wieloznaczne (”+”,”#”).

Przykłady:

```
path/to/file
path/to/#
path/+/topic
```

Plugins.MQTT.Sessions.<SessionName>.Url

Określa ciąg połączenia z brokerem MQTT dla nazwanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Nie powinien zawierać parametrów zapytania.
 Musi odpowiadać formatowi URL.
 Obsługiwane schematy: `tcp` (domyślnie), `ws`, `tls`
 Schemat może zostać pominięty.
 Port może zostać pominięty (domyślnie=1883)

Domyślnie: `tcp://localhost:1883`

Przykłady:

```
tcp://host:1883
localhost
ws://host:8080
```

Plugins.MQTT.Sessions.<SessionName>.User

Określa nazwę użytkownika dla nazwanej sesji używanej do łączenia z MQTT.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Jeśli musisz ustawić limit czasu wykonania żądania (jak długo czekać na zakończenie żądania przed jego zamknięciem), użyj formularza konfiguracji pozycji.

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka MSSQL

Omówienie

Parametry konfiguracyjne umożliwiają dostosowanie zachowania wtyczki MSSQL.

Zaleca się określanie ich we własnym pliku konfiguracyjnym (np. `mssql.conf`), a nie w pliku konfiguracyjnym agent 2. Następnie należy użyć dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wtyczka MSSQL jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki MSSQL.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Omówienie parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.MSSQL.CustomQueriesPath	Określa ścieżkę do katalogu zawierającego zdefiniowane przez użytkownika pliki `.sql` z niestandardowymi zapytaniami, które wtyczka może wykonywać.
Plugins.MSSQL.CustomQueriesEnabled	Włącza wykonywanie klucza pozycja `mssql.custom.query`.
Plugins.MSSQL.Default.CACertPath	Określa domyślną ścieżkę do pliku certyfikatu klucza publicznego urzędu certyfikacji (CA), który wystawił certyfikat serwera MSSQL.
Plugins.MSSQL.Default.Database	Określa domyślną nazwę bazy danych.

Parameter	Description
Plugins.MSSQL.Default.Encrypt	Określa domyślny poziom szyfrowania.
Plugins.MSSQL.Default.HostNameInCertificate	Określa domyślną nazwę pospolitą (CN) certyfikatu serwera MSSQL.
Plugins.MSSQL.Default.Password	Określa domyślne hasło.
Plugins.MSSQL.Default.TLSMinVersion	Określa domyślną minimalną wersję TLS.
Plugins.MSSQL.Default.TrustServerCertificate	Domyślnie ufa certyfikatowi serwera bez jego weryfikacji.
Plugins.MSSQL.Default.Uri	Określa domyślny URI.
Plugins.MSSQL.Default.User	Określa domyślną nazwę użytkownika.
Plugins.MSSQL.KeepAlive	Określa maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.MSSQL.Sessions.<SessionName>.CACertPath	Określa ścieżkę do pliku sesji o podanej nazwie zawierającego certyfikat klucza publicznego urzędu certyfikacji (CA), który wystawił certyfikat serwera MSSQL.
Plugins.MSSQL.Sessions.<SessionName>.Database	Określa nazwę bazy danych dla sesji o podanej nazwie.
Plugins.MSSQL.Sessions.<SessionName>.Encrypt	Określa poziom szyfrowania dla sesji o podanej nazwie.
Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate	Określa nazwę pospolitą (CN) certyfikatu serwera MSSQL dla sesji o podanej nazwie.
Plugins.MSSQL.Sessions.<SessionName>.Password	Określa hasło dla sesji o podanej nazwie.
Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion	Określa minimalną wersję TLS dla sesji o podanej nazwie.
Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate	Domyślnie ufa certyfikatowi serwera dla sesji o podanej nazwie bez jego weryfikacji.
Plugins.MSSQL.Sessions.<SessionName>.Uri	Określa URI dla sesji o podanej nazwie.
Plugins.MSSQL.Sessions.<SessionName>.User	Określa nazwę użytkownika dla sesji o podanej nazwie.
Plugins.MSSQL.System.Path	Określa ścieżkę do pliku wykonywalnego wtyczki MSSQL.
Plugins.MSSQL.Timeout	Ustawia czas oczekiwania, w sekundach, na odpowiedź serwera podczas pierwszego połączenia oraz podczas kolejnych operacji w sesji.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.MSSQL.CustomQueriesDir

Określa ścieżkę do katalogu zawierającego zdefiniowane przez użytkownika pliki `.sql` z niestandardowymi zapytaniami, które plugin może wykonywać.
 Plugin ładuje wszystkie dostępne pliki .sql w skonfigurowanym katalogu podczas uruchamiania. Oznacza to, że wszelkie zmiany w plikach niestandardowych zapytań nie zostaną uwzględnione, dopóki plugin nie zostanie ponownie uruchomiony. Plugin jest uruchamiany i zatrzymywany razem z agentem Zabbix agent 2.

Wartości domyślne:
`/usr/local/share/zabbix/custom-queries/mssql` - dla systemów Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\MSSQL` - dla systemów Windows (gdzie `*` to nazwa dysku z zmiennej środowiskowej `ProgramFiles`)

Plugins.MSSQL.CustomQueriesEnabled

Włącz wykonywanie klucza pozycja `mssql.custom.query`.
 Jeśli opcja jest wyłączona, żadne zapytania nie zostaną załadowane z katalogu niestandardowych zapytań.

Domyślnie: `false`
 Zakres: `true/false`

Plugins.MSSQL.Default.CACertPath

Określ domyślną ścieżkę do pliku z certyfikatem klucza publicznego urzędu certyfikacji (CA), który wydał certyfikat serwera MSSQL.
 Certyfikat musi być w formacie PEM.

Plugins.MSSQL.Default.Database

Określ domyślną nazwę bazy danych.

Plugins.MSSQL.Default.Encrypt

Określa poziom domyślnego szyfrowania.

Zakres:
true - przesyłanie danych między pluginem a serwerem jest szyfrowane;
false - przesyłanie danych między pluginem a serwerem nie jest szyfrowane poza pakietem logowania;
strict - przesyłanie danych między pluginem a serwerem jest szyfrowane end-to-end z użyciem TDS8;
disable - przesyłanie danych między pluginem a serwerem nie jest szyfrowane.

Plugins.MSSQL.Default.HostNameInCertificate

Określ domyślną nazwę pospolitą (CN) certyfikatu serwera MSSQL.

Plugins.MSSQL.Default.Password

Określ domyślne hasło dla chronionego serwera MSSQL.

Plugins.MSSQL.Default.TLSMinVersion

Określ domyślną minimalną wersję TLS.

Zakres: `1.0`, `1.1`, `1.2`, `1.3`

Plugins.MSSQL.Default.TrustServerCertificate

Domyślnie ufaj certyfikatowi serwera bez jego weryfikacji.

Zakres: `true`, `false`

Plugins.MSSQL.Default.Uri

Określ domyślny URI do łączenia się z serwerem MSSQL.
 Osadzone poświadczenia zostaną zignorowane.
 Jedynym obsługiwanym schematem jest `sqlserver://`.
 Schemat można pominąć.

Domyślnie: `sqlserver://localhost:1433`

Plugins.MSSQL.Default.User

Określ domyślną nazwę użytkownika dla chronionego serwera MSSQL.

Plugins.MSSQL.KeepAlive

Określa maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60–900

Plugins.MSSQL.Sessions.<SessionName>.CACertPath

Określ ścieżkę do pliku sesji o podanej nazwie do publicznego certyfikatu klucza urzędu certyfikacji (CA), który wystawił certyfikat serwera MSSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Certyfikat musi być w formacie PEM.

Plugins.MSSQL.Sessions.<SessionName>.Database

Określ nazwę bazy danych dla wskazanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MSSQL.Sessions.<SessionName>.Encrypt

Określa poziom szyfrowania nazwanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres:
true - przesyłanie danych między pluginem a serwerem jest szyfrowane;
false - przesyłanie danych między pluginem a serwerem nie jest szyfrowane poza pakietem logowania;
strict - przesyłanie danych między pluginem a serwerem jest szyfrowane end-to-end z użyciem TDS8;
disable - przesyłanie danych między pluginem a serwerem nie jest szyfrowane.

Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate

Określ nazwę pospolitą (CN) certyfikatu serwera MSSQL dla nazwanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MSSQL.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji do chronionego serwera MSSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion

Określa minimalną wersję TLS dla wskazanej sesji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres: `1.0`, `1.1`, `1.2`, `1.3`

Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate

Zaufaj certyfikatowi serwera dla wskazanej sesji bez jego weryfikacji.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres: `true`, `false`

Plugins.MSSQL.Sessions.<SessionName>.Uri

Określ URI nazwanej sesji do łączenia się z serwerem MSSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Osadzone poświadczenia zostaną zignorowane.
 Jedynym obsługiwanym schematem jest `sqlserver://`.
 Schemat można pominąć.

Domyślnie: `sqlserver://localhost:1433`

Plugins.MSSQL.Sessions.<SessionName>.User

Określ nazwę użytkownika dla nazwanej sesji na chronionym serwerze MSSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.MSSQL.System.Path

Określ ścieżkę do pliku wykonywalnego wtyczki MSSQL.
 Globalne ustawienie dla wtyczki MSSQL. Ma zastosowanie do wszystkich połączeń.

Przykład: `Plugins.MSSQL.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-mssql`

Plugins.MSSQL.Timeout

Ustaw czas oczekiwania, w sekundach, na odpowiedź serwera podczas pierwszego połączenia oraz podczas kolejnych operacji w sesji.

Domyślnie: `global timeout`
 Zakres: 1-30

Opcje

Parameter	Description
-V --version	Wyświetla wersję wtyczki i informacje o licencji.
-h --help	Wyświetla informacje pomocy (skrót).
-t, --test <item key>	Uruchamia wtyczkę do testowania (konfiguracja wtyczki jest ignorowana).

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka MySQL

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania wtyczki MySQL.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `mysql.conf`), a nie w pliku konfiguracji agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Mysql.CallTimeout	Określa maksymalny czas w sekundach oczekiwania na zakończenie żądania.
Plugins.Mysql.CustomQueriesPath	Określa pełną ścieżkę do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami. Włącza wykonywanie klucza pozycja `mysql.custom.query`.
Plugins.Mysql.CustomQueriesEnabled	
Plugins.Mysql.Default.Password	Określa domyślne hasło.
Plugins.Mysql.Default.TLSConnect	Określa domyślny poziom szyfrowania komunikacji między Zabbix agent 2 a monitorowanymi bazami danych.
Plugins.Mysql.Default.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer.
Plugins.Mysql.Default.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów.
Plugins.Mysql.Default.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych.
Plugins.Mysql.Default.Uri	Określa domyślny URI.
Plugins.Mysql.Default.User	Określa domyślną nazwę użytkownika.

Parameter	Description
Plugins.Mysql.KeepAlive	Określa maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Mysql.Sessions.<SessionName>.Password	Określa hasło dla nazwanej sesji.
Plugins.Mysql.Sessions.<SessionName>.TLSConnect	Określa poziom szyfrowania dla nazwanej sesji komunikacji między Zabbix agent 2 a monitorowanymi bazami danych.
Plugins.Mysql.Sessions.<SessionName>.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer dla nazwanej sesji.
Plugins.Mysql.Sessions.<SessionName>.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów dla nazwanej sesji.
Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych dla nazwanej sesji.
Plugins.Mysql.Sessions.<SessionName>.Uri	Określa URI nazwanej sesji.
Plugins.Mysql.Sessions.<SessionName>.User	Określa nazwę użytkownika nazwanej sesji.
Plugins.Mysql.Timeout	Ustawia maksymalny czas oczekiwania w sekundach podczas nawiązywania połączenia.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Mysql.CallTimeout

Określa maksymalny czas oczekiwania w sekundach na zakończenie żądania.

Domyślnie: `global timeout`
 Zakres: 1-30

Plugins.Mysql.CustomQueriesPath

Określ pełną ścieżkę do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.

Wartości domyślne:
`/usr/local/share/zabbix/custom-queries/mysql` - dla systemów Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\Mysql` - dla systemów Windows (gdzie `*` jest nazwą dysku z zmiennej środowiskowej `ProgramFiles`)

Plugins.Mysql.CustomQueriesEnabled

Włącz wykonywanie klucza pozycja `mysql.custom.query`.
 Jeśli opcja jest wyłączona, żadne zapytania nie będą wczytywane z katalogu niestandardowych zapytań.

Domyślnie: `false`
 Zakres: `true`/`false`

Plugins.Mysql.Default.Password

Określ domyślne hasło dla chronionego serwera MySQL.

Plugins.Mysql.Default.TLSConnect

Określ domyślny poziom szyfrowania komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych.

Zakres:
required - wymagaj połączenia TLS;
verify_ca - weryfikuj certyfikaty;
verify_full - weryfikuj certyfikaty i adres IP

Plugins.Mysql.Default.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) do weryfikacji certyfikatu peer, używaną w szyfrowanej komunikacji między agentem 2 Zabbix a monitorowanymi bazami danych.

Obowiązkowe: `yes`, jeśli `Plugins.Mysql.Default.TLSConnect` jest ustawione na `verify_ca` lub `verify_full`

Plugins.Mysql.Default.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agent lub łańcuch certyfikatów, używany do szyfrowanej komunikacji między agent 2 Zabbix a monitorowanymi bazami danych.

Obowiązkowe: `yes`, jeśli określono `Plugins.Mysql.Default.TLSKeyFile`

Plugins.Mysql.Default.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych, używany do szyfrowanej komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych.

Wymagane: yes, jeśli określono Plugins.Mysql.Default.TLSCertFile

Plugins.Mysql.Default.Uri

Określ domyślny URI do łączenia się z serwerem MySQL.
 Osadzone poświadczenia zostaną zignorowane.
 Obsługiwane schematy: tcp, unix
 Schemat można pominąć.
 Port można pominąć (domyślnie=3306).

Przykłady:

```
tcp://localhost:3306
localhost
unix:/var/run/mysql.sock
```

Plugins.Mysql.Default.User

Określ domyślną nazwę użytkownika dla chronionego serwera MySQL.

Plugins.Mysql.KeepAlive

Określ maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60–900

Plugins.Mysql.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji do chronionego serwera MySQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Mysql.Sessions.<SessionName>.TLSConnect

Określa poziom szyfrowania dla nazwanej sesji komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres:
required - wymagaj połączenia TLS;
verify_ca - weryfikuj certyfikaty;
verify_full - weryfikuj certyfikaty i adres IP

Plugins.Mysql.Sessions.<SessionName>.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA) do weryfikacji certyfikatu peer dla nazwanej sesji, używaną do szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.

Wymagane: yes, jeśli Plugins.Mysql.Sessions.<SessionName>.TLSConnect jest ustawione na verify_ca lub verify_full

Plugins.Mysql.Sessions.<SessionName>.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agenta lub łańcuch certyfikatów dla wskazanej sesji, używany do szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Wymagane: yes, jeśli określono Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile

Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz bazy danych dla wskazanej sesji, używany do szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Wymagane: yes, jeśli określono Plugins.Mysql.Sessions.<SessionName>.TLSCertFile

Plugins.Mysql.Sessions.<SessionName>.Uri

Określ URI nazwanej sesji do łączenia się z serwerem MySQL.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
 Osadzone poświadczenia zostaną zignorowane.
 Obsługiwane schematy: tcp, unix
 Schemat można pominąć.
 Port można pominąć (domyślnie=3306).

Przykłady:

```
tcp://localhost:3306
localhost
unix:/var/run/mysql.sock
```

Plugins.Mysql.Sessions.<SessionName>.User

Określ nazwę użytkownika dla nazwanej sesji na chronionym serwerze MySQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Mysql.Timeout

Ustaw maksymalny czas oczekiwania na nawiązanie połączenia, w sekundach.

Domyślnie: `global timeout<br>` Zakres: 1-30

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje dotyczące konfiguracji wtyczek

Wtyczka NVIDIA GPU

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania wtyczki NVIDIA GPU.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `nvidiagpu.conf`), a nie w pliku konfiguracji agenta 2. Następnie należy użyć dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wtyczka NVIDIA GPU jest wtyczką ładowaną dynamicznie i jest dostępna oraz w pełni opisana w repozytorium wtyczki NVIDIA GPU.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.NVIDIA.System.Path	Określa ścieżkę do pliku wykonywalnego wtyczki NVIDIA GPU.
Plugins.NVIDIA.Timeout	Określa maksymalny czas (w sekundach) oczekiwania na odpowiedź serwera podczas prób połączenia i kolejnych operacji w sesji.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.NVIDIA.System.Path

Określ ścieżkę do pliku wykonywalnego wtyczki NVIDIA GPU.

Przykład:

```
Plugins.NVIDIA.System.Path=/path/to/executable/nvidia
```

Plugins.NVIDIA.Timeout

Określa maksymalny czas (w sekundach) oczekiwania na odpowiedź serwera podczas prób połączenia oraz kolejnych operacji w sesji.
 Globalny limit czasu dla typu pozycji lub indywidualny limit czasu pozycji zastąpi tę wartość, jeśli będzie większy.

Domyślnie: `global timeout<br>` Zakres: 1-30

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka Oracle

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania pluginu Oracle.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `oracle.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie należy użyć dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Oracle.CallTimeout	Maksymalny czas oczekiwania w sekundach na zakończenie żądania. Ten parametr obecnie nie ma wpływu; zamiast niego używane jest ustawienie item timeout.
Plugins.Oracle.ConnectTimeout	Określa maksymalny czas oczekiwania w sekundach na nawiązanie połączenia.
Plugins.Oracle.CustomQueriesPath	Określa pełną ścieżkę do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.
Plugins.Oracle.CustomQueriesEnabled	Włącza wykonywanie klucza pozycja `oracle.custom.query`.
Plugins.Oracle.Default.Password	Określa domyślne hasło.
Plugins.Oracle.Default.Service	Określa domyślną nazwę usługi.
Plugins.Oracle.Default.URI	Określa domyślny URI.
Plugins.Oracle.Default.User	Określa domyślną nazwę użytkownika.
Plugins.Oracle.KeepAlive	Ustawia maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń pluginu.
Plugins.Oracle.ResolveTNS	Określa, jak interpretować ciąg połączenia (ConnString) podczas łączenia z serwerem Oracle.
Plugins.Oracle.Sessions.<SessionName>.Password	Określa hasło dla nazwanej sesji.
Plugins.Oracle.Sessions.<SessionName>.Service	Określa nazwę usługi dla nazwanej sesji.
Plugins.Oracle.Sessions.<SessionName>.URI	Określa URI dla nazwanej sesji.
Plugins.Oracle.Sessions.<SessionName>.User	Określa nazwę użytkownika dla nazwanej sesji.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Oracle.CallTimeout

Maksymalny czas oczekiwania w sekundach na zakończenie żądania.
 Ten parametr obecnie nie ma żadnego wpływu. Maksymalny czas oczekiwania jest zamiast tego określany przez item timeout.

Domyślnie: `Timeout`[1]
 Zakres: 1-30

Plugins.Oracle.ConnectTimeout

Określa maksymalny czas oczekiwania w sekundach na nawiązanie połączenia.

Domyślnie: `Timeout`[1]
 Zakres: 1-30

Plugins.Oracle.CustomQueriesPath

Określ pełną ścieżkę do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.

Wartości domyślne:
`/usr/local/share/zabbix/custom-queries/oracle` - dla systemów Unix,
`*:\Program Files\Zabbix Agent 2\Custom Queries\Oracle` - dla systemów Windows (gdzie `*` to nazwa dysku z zmiennej środowiskowej `ProgramFiles`)

Przykład:

```
/etc/zabbix/oracle/sql
```

Wtyczki.Oracle.CustomQueriesEnabled

Włącza wykonywanie klucza pozycja `oracle.custom.query`.
 Jeśli wyłączone, żadne zapytania nie będą ładowane z katalogu niestandardowych zapytań.

Domyślnie: `false`
 Zakres: `true/false`

Plugins.Oracle.Default.Password

Określ domyślne hasło do łączenia z Oracle.
 Używane, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.

Plugins.Oracle.Default.Service

Określa domyślną nazwę usługi do łączenia z Oracle (SID nie jest obsługiwany).
 Używane, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.

Plugins.Oracle.Default.Uri

Określa domyślny URI do łączenia z Oracle.
 Używany, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.
 Osadzone poświadczenia zostaną zignorowane.
 Musi odpowiadać formatowi URI.
 Obsługiwane schematy: `tcp`, `tcps`
 Schemat można pominąć.
 Port można pominąć (domyślnie=1521).
 Można również określić klucz TNS lub wartość TNS jako ciąg połączenia. Wartość TNS musi być zbudowana bez białych znaków.

Przykłady:

```
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

Określ domyślną nazwę użytkownika do łączenia z Oracle.
 Używana, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.

Plugins.Oracle.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń z wtyczką.

Domyślnie: 300
 Zakres: 60-900

Plugins.Oracle.ResolveTNS

Określa, jak interpretować ciąg połączenia (ConnString) podczas łączenia z serwerem Oracle.
 Jeśli ustawiono `false`, domyślny schemat i port zostaną dodane do ConnString (jeśli ich brakuje).
 Jeśli opcja jest ustawiona na `true`, domyślny schemat i port zostaną pominięte (chyba że zostały jawnie określone w ConnString), a ConnString zostanie przekazany do klienta Oracle bez zmian. Jeśli klient Oracle znajdzie ConnString w pliku `tnsnames.ora`, do połączenia z serwerem Oracle zostanie użyty znaleziony opis połączenia.

Domyślnie: `true`
 Zakres: `true`/`false`

Plugins.Oracle.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji używane do połączenia z Oracle.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Oracle.Sessions.<SessionName>.Service

Określ nazwę usługi dla nazwanej sesji używanej do łączenia z Oracle (SID nie jest obsługiwany).
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.Oracle.Sessions.<SessionName>.Uri

Określ URI nazwanej sesji do łączenia z Oracle.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
 Osadzone poświadczenia zostaną zignorowane.
 Musi odpowiadać formatowi URI.
 Obsługiwane schematy: `tcp`, `tcps`
 Schemat może zostać pominięty.
 Port może zostać pominięty (domyślnie=1521).
 Możliwe jest również określenie klucza TNS lub wartości TNS jako ciągu połączenia. Wartość TNS musi być złożona bez białych znaków.

Przykłady:

```
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

Określ nazwę użytkownika dla nazwanej sesji używanej do łączenia z Oracle.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Przypisy

[1] Limit czasu jest definiowany przez opcję `Timeout` w pliku konfiguracyjnym (`zabbix_agent2.conf`) Zabbix agent 2 (UNIX) lub Zabbix agent 2 (Windows).

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka PostgreSQL

Przegląd

Parametry konfiguracyjne umożliwiają dostosowanie działania wtyczki PostgreSQL.

Zaleca się określanie ich w osobnym pliku konfiguracyjnym (np. `postgresql.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie należy użyć dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wtyczka PostgreSQL jest wtyczką ładowaną i jest dostępna oraz w pełni opisana w repozytorium wtyczki PostgreSQL.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Przegląd parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.PostgreSQL.CallTimeout	Ustaw maksymalny czas oczekiwania w sekundach na zakończenie żądania.
Plugins.PostgreSQL.CustomQueriesPath	Określ pełną ścieżkę do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.
Plugins.PostgreSQL.CustomQueriesEnabled	Włącz wykonywanie klucza pozycja `postgresql.custom.query`.
Plugins.PostgreSQL.Default.CacheMode	Określ domyślny tryb pamięci podręcznej dla połączenia PostgreSQL.
Plugins.PostgreSQL.Default.Database	Określ domyślną nazwę bazy danych.
Plugins.PostgreSQL.Default.Password	Określ domyślne hasło.
Plugins.PostgreSQL.Default.TLSCAFile	Określ pełną ścieżkę do pliku zawierającego certyfikat(y) głównego urzędu CA do weryfikacji certyfikatu peer.
Plugins.PostgreSQL.Default.TLSCertFile	Określ pełną ścieżkę do pliku zawierającego certyfikat PostgreSQL lub łańcuch certyfikatów.
Plugins.PostgreSQL.Default.TLSMode	Określ domyślny poziom szyfrowania komunikacji między Zabbix agent 2 a monitorowanymi bazami danych.
Plugins.PostgreSQL.Default.TLSKeyFile	Określ pełną ścieżkę do pliku zawierającego prywatny klucz PostgreSQL.
Plugins.PostgreSQL.Default.Uri	Określ domyślny URI.
Plugins.PostgreSQL.Default.User	Określ domyślną nazwę użytkownika.
Plugins.PostgreSQL.KeepAlive	Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.PostgreSQL.Sessions.<SessionName>.CacheMode	Określ nazwany tryb pamięci podręcznej sesji dla połączenia PostgreSQL.
Plugins.PostgreSQL.Sessions.<SessionName>.Database	Określ nazwę bazy danych dla nazwanej sesji.
Plugins.PostgreSQL.Sessions.<SessionName>.Password	Określ hasło dla nazwanej sesji.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile	Określ pełną ścieżkę do pliku zawierającego certyfikat(y) głównego urzędu CA do weryfikacji certyfikatu peer.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile	Określ pełną ścieżkę do pliku zawierającego certyfikat PostgreSQL lub łańcuch certyfikatów.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSMode	Określ poziom szyfrowania dla nazwanej sesji w komunikacji między Zabbix agent 2 a monitorowanymi bazami danych.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile	Określ pełną ścieżkę do pliku zawierającego prywatny klucz PostgreSQL.
Plugins.PostgreSQL.Sessions.<SessionName>.Uri	Określ URI dla nazwanej sesji.
Plugins.PostgreSQL.Sessions.<SessionName>.User	Określ nazwę użytkownika dla nazwanej sesji.

|Plugins.PostgreSQL.System.Path|Określ ścieżkę do pliku wykonywalnego wtyczki PostgreSQL.| |Plugins.PostgreSQL.Timeout|Ustaw limit czasu wykonania żądania.|

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.PostgreSQL.CallTimeout

Ustaw maksymalny czas oczekiwania w sekundach na zakończenie żądania.

Domyślnie: `global timeout`
 Zakres: 1-30

Plugins.PostgreSQL.CustomQueriesPath

Określ pełną ścieżkę do katalogu zawierającego pliki `.sql` z niestandardowymi zapytaniami.

Domyślnie:
/usr/local/share/zabbix/custom-queries/postgreSQL - dla systemów Unix,
*:\Program Files\Zabbix Agent 2\Custom Queries\PostgreSQL - dla systemów Windows (gdzie * to nazwa dysku z zmiennej środowiskowej ProgramFiles)

Przykład:

```
/etc/zabbix/oracle/sql
```

Plugins.PostgreSQL.CustomQueriesEnabled

Włącz wykonywanie klucza pozycja postgresql.custom.query.
 Jeśli opcja jest wyłączona, żadne zapytania nie będą ładowane z katalogu niestandardowych zapytań.

Domyślnie: false
 Zakres: true/false

Plugins.PostgreSQL.Default.CacheMode

Określa domyślny tryb pamięci podręcznej dla połączenia PostgreSQL.

Domyślnie: prepare
 Zakres:
prepare - tworzy przygotowane instrukcje na serwerze PostgreSQL;
describe - używa anonimowej przygotowanej instrukcji do opisania instrukcji bez tworzenia instrukcji na serwerze.

Należy pamiętać, że describe jest przydatne przede wszystkim wtedy, gdy środowisko nie pozwala na używanie przygotowanych instrukcji, na przykład podczas korzystania z puli połączeń, takiej jak PgBouncer.

Plugins.PostgreSQL.Default.Database

Określ domyślną bazę danych do połączenia z PostgreSQL.
 Używana, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.

Plugins.PostgreSQL.Default.Password

Określ domyślne hasło do łączenia się z PostgreSQL.
 Używane, jeśli nie podano wartości w kluczu pozycji lub nazwanej sesji.

Plugins.PostgreSQL.Default.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikat(y) głównego urzędu CA do weryfikacji certyfikatu peer w szyfrowanej komunikacji między agentem 2 Zabbix a monitorowanymi bazami danych.
 Używane, jeśli w nazwanej sesji nie określono żadnej wartości.

Wymagane: yes, jeśli Plugins.PostgreSQL.Default.TLSConnect ma ustawioną wartość verify_ca lub verify_full

Plugins.PostgreSQL.Default.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat PostgreSQL lub łańcuch certyfikatów do szyfrowanej komunikacji między agent 2 Zabbix a monitorowanymi bazami danych.
 Używane, jeśli nie określono wartości w nazwanej sesji.

Obowiązkowe: yes, jeśli określono Plugins.PostgreSQL.Default.TLSKeyFile

Plugins.PostgreSQL.Default.TLSConnect

Określ domyślny poziom szyfrowania komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych.
 Używane, jeśli w nazwanej sesji nie określono żadnej wartości.

Zakres:
required - połącz przy użyciu TLS jako trybu transportu bez sprawdzania tożsamości;
verify_ca - połącz przy użyciu TLS i zweryfikuj certyfikat;
verify_full - połącz przy użyciu TLS, zweryfikuj certyfikat i sprawdź, czy tożsamość bazy danych (CN) określona przez DBHost zgadza się z jej certyfikatem.

Nieokreślony typ szyfrowania oznacza połączenie nieszyfrowane.

Plugins.PostgreSQL.Default.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz PostgreSQL do szyfrowanej komunikacji między agent 2 Zabbix a monitorowanymi bazami danych.
 Używane, jeśli nie określono wartości w nazwanej sesji.

Obowiązkowe: yes, jeśli określono Plugins.PostgreSQL.Default.TLSCertFile

Plugins.PostgreSQL.Default.Uri

Określ domyślny URI do łączenia z PostgreSQL.
 Używany, jeśli w kluczu pozycja lub nazwanej sesji nie określono żadnej wartości.
 Osadzone poświadczenia zostaną zignorowane.
 Musi odpowiadać formatowi URI.
 Obsługiwane schematy: tcp, unix

Przykłady:

```
tcp://127.0.0.1:5432
tcp://localhost
unix:/var/run/postgresql/.s.PGSQL.5432
```

Plugins.PostgreSQL.Default.User

Określ domyślną nazwę użytkownika do łączenia z PostgreSQL.
 Używana, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.

Plugins.PostgreSQL.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Plugins.PostgreSQL.Sessions.<SessionName>.CacheMode

Określa nazwany tryb pamięci podręcznej sesji dla połączenia PostgreSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Domyślnie: `prepare`
 Zakres:
`prepare` - tworzy przygotowane instrukcje na serwerze PostgreSQL;
`describe` - używa anonimowej przygotowanej instrukcji do opisania instrukcji bez tworzenia instrukcji na serwerze.

Należy pamiętać, że `describe` jest przydatne przede wszystkim wtedy, gdy środowisko nie zezwala na przygotowane instrukcje, na przykład podczas korzystania z poolera połączeń, takiego jak PgBouncer.

Plugins.PostgreSQL.Sessions.<SessionName>.Database

Określ nazwę bazy danych sesji dla połączenia z PostgreSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.PostgreSQL.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji używane do łączenia z PostgreSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.
 Musi odpowiadać formatowi hasła.

Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikat(y) głównego urzędu CA do weryfikacji certyfikatu peer w szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Wymagane: `yes`, jeśli `Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect` jest ustawione na `verify_ca` lub `verify_full`

Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat PostgreSQL lub łańcuch certyfikatów dla szyfrowanej komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.

Wymagane: `yes`, jeśli określono `Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile`

Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect

Określa nazwany poziom szyfrowania sesji dla komunikacji między agentem Zabbix 2 a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Zakres:
 `required` - łączy się przy użyciu TLS jako trybu transportu bez sprawdzania tożsamości;
 `verify_ca` - łączy się przy użyciu TLS i weryfikuje certyfikat;
 `verify_full` - łączy się przy użyciu TLS, weryfikuje certyfikat oraz sprawdza, czy tożsamość bazy danych (CN) określona przez DBHost jest zgodna z jej certyfikatem.

Nieokreślony typ szyfrowania oznacza połączenie nieszyfrowane.

Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz PostgreSQL do szyfrowanej komunikacji między agentem Zabbix agent 2 a monitorowanymi bazami danych.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.

Obowiązkowe: `yes`, jeśli określono `Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile`.

Plugins.PostgreSQL.Sessions.<SessionName>.Uri

Określ URI nazwanej sesji do łączenia z PostgreSQL.
 <SessionName> - zdefiniuj nazwę sesji do użycia w kluczach pozycji.
 Osadzone poświadczenia zostaną zignorowane.
 Musi odpowiadać formatowi URI.
 Obsługiwane schematy: `tcp`, `unix`

Przykłady:

```
tcp://127.0.0.1:5432
tcp://localhost
unix:/var/run/postgresql/.s.PGSQL.5432
```

Plugins.PostgreSQL.Sessions.<SessionName>.User

Określ nazwę użytkownika dla wskazanej sesji, używaną do połączenia z PostgreSQL.
 <SessionName> - określa nazwę sesji do użycia w kluczach pozycji.

Plugins.PostgreSQL.System.Path

Określ ścieżkę do pliku wykonywalnego wtyczki PostgreSQL.

Obowiązkowe: `yes`

Przykład:

```
Plugins.PostgreSQL.System.Path=/usr/libexec/zabbix/zabbix-agent2-plugin-postgresql
```

Plugins.PostgreSQL.Timeout

Ustawia limit czasu wykonania żądania (czas w sekundach, przez jaki należy czekać na zakończenie żądania przed jego przerwaniem).

Domyślnie: `global timeout`
 Zakres: 1-30

Opcje

Parameter	Description
-V --version	Wyświetl wersję wtyczki oraz informacje o licencji.
-h --help	Wyświetl informacje pomocy (skrót).
-t, --test <item key>	Uruchom wtyczkę do testowania (konfiguracja wtyczki jest ignorowana).

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka Redis

Omówienie

Parametry konfiguracyjne umożliwiają dostosowanie działania wtyczki Redis.

Zaleca się określanie ich we własnym pliku konfiguracyjnym (np. `redis.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Omówienie parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Redis.Default.Password	Określa domyślne hasło do łączenia z Redis.
Plugins.Redis.Default.TLSConnect	Określa poziom szyfrowania wymagany dla komunikacji Zabbix agent 2.
Plugins.Redis.Default.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA.
Plugins.Redis.Default.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta.
Plugins.Redis.Default.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz agenta.
Plugins.Redis.Default.Uri	Określa domyślny URI do łączenia z Redis.
Plugins.Redis.Default.User	Określa domyślną nazwę użytkownika do łączenia z Redis.
Plugins.Redis.KeepAlive	Ustawia maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.
Plugins.Redis.Sessions.<SessionName>.Password	Określa hasło dla nazwanej sesji do łączenia z Redis.
Plugins.Redis.Sessions.<SessionName>.TLSConnect	Określa poziom szyfrowania wymagany dla komunikacji Zabbix agent 2 dla nazwanej sesji.
Plugins.Redis.Sessions.<SessionName>.TLSCAFile	Określa pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA dla nazwanej sesji.
Plugins.Redis.Sessions.<SessionName>.TLSCertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat agenta dla nazwanej sesji.
Plugins.Redis.Sessions.<SessionName>.TLSKeyFile	Określa pełną ścieżkę do pliku zawierającego prywatny klucz agenta dla nazwanej sesji.
Plugins.Redis.Sessions.<SessionName>.Uri	Określa URI dla nazwanej sesji do łączenia z Redis.

Parameter	Description
Plugins.Redis.Sessions.<SessionName>.User	Określa nazwę użytkownika nazwanej sesji do łączenia z Redis.
Plugins.Redis.Timeout	Ustawia limit czasu wykonania żądania.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM.
- Komentarze zaczynające się od ”#” są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Redis.Default.Password

Określ domyślne hasło do łączenia z Redisem.
 Używane, jeśli nie podano wartości w kluczu pozycji lub nazwanej sesji.

Plugins.Redis.Default.TLSConnect

Określ poziom szyfrowania wymagany dla komunikacji Zabbix agent 2.
 Pusta lub nieustawiona wartość jest traktowana jako brak TLS.

Zakres: `required` - szyfrowane, niezweryfikowane (tylko do testów), `verify_ca` - szyfrowane, certyfikat serwera, `verify_full` - szyfrowane, certyfikat serwera i nazwa serwera zweryfikowane z użyciem SAN

Plugins.Redis.Default.TLSCAFile

Podaj pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów certyfikacji (CA).

Plugins.Redis.Default.TLSCertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat agenta.

Plugins.Redis.Default.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz agenta.

Plugins.Redis.Default.Uri

Określa domyślny URI do łączenia z Redis.
 Używany, jeśli nie określono wartości w kluczu pozycji lub nazwanej sesji.
 Osadzone poświadczenia zostaną zignorowane. Musi odpowiadać formatowi URI.
 Obsługiwane schematy: `tcp`, `unix`
 Schemat można pominąć.
 Port można pominąć (domyślnie=6379).

Domyślnie: `tcp://localhost:6379`

Przykłady:

```
tcp://localhost:6379
localhost
unix:/var/run/redis.sock
```

Plugins.Redis.Default.User

Określa domyślną nazwę użytkownika do łączenia z Redis.
 Używane, jeśli w kluczu pozycji lub nazwanej sesji nie określono żadnej wartości.

Domyślnie: `default`

Plugins.Redis.KeepAlive

Ustaw maksymalny czas oczekiwania (w sekundach) przed zamknięciem nieużywanych połączeń wtyczki.

Domyślnie: 300
 Zakres: 60–900

Plugins.Redis.Sessions.<SessionName>.Password

Określ hasło dla nazwanej sesji używane do połączenia z Redis.
 <SessionName> - zdefiniuj nazwę sesji, która ma być używana w kluczach pozycji.

Plugins.Redis.Sessions.<SessionName>.TLSConnect

Określ poziom szyfrowania wymagany dla komunikacji Zabbix agent 2 dla nazwanej sesji.
 <SessionName> - określa nazwę sesji używaną w kluczach pozycji.
 Pusta lub nieustawiona wartość jest traktowana jako brak TLS.

Zakres: `required` - szyfrowane, niezweryfikowane (tylko do testów), `verify_ca` - szyfrowane, certyfikat serwera, `verify_full` - szyfrowane, certyfikat serwera i nazwa serwera zweryfikowane za pomocą SAN

Plugins.Redis.Sessions.<SessionName>.TLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikaty głównych urzędów CA dla wskazanej sesji.
 <SessionName> - określa nazwę sesji, która ma być używana w kluczach pozycji.

Obowiązkowe: `yes`, jeśli `Plugins.Redis.Sessions.<SessionName>.TLSConnect` ma ustawioną wartość `verify_ca` lub `verify_full`

Plugins.Redis.Sessions.<SessionName>.TLSCertFile

Podaj pełną ścieżkę do pliku zawierającego certyfikat agenta dla nazwanego sesji.
 <SessionName> - określa nazwę sesji, która ma być używana w kluczach pozycji.

Plugins.Redis.Sessions.<SessionName>.TLSKeyFile

Określ pełną ścieżkę do pliku zawierającego prywatny klucz agenta dla wskazanej sesji.
 <SessionName> - określa nazwę sesji, która ma być używana w kluczach pozycji.

Plugins.Redis.Sessions.<SessionName>.Uri

Określa URI nazwanej sesji do połączenia z Redis.
 <SessionName> - określa nazwę sesji, która ma być używana w kluczach pozycji.
 Osadzone poświadczenia zostaną zignorowane. Musi odpowiadać formatowi URI.
 Obsługiwane schematy: `tcp`, `unix`
 Schemat może zostać pominięty.
 Port może zostać pominięty (domyślnie=6379).

Domyślnie: `tcp://localhost:6379`

Przykłady:

```
tcp://localhost:6379
localhost
unix:/var/run/redis.sock
```

Plugins.Redis.Sessions.<SessionName>.User

Określ nazwę użytkownika dla wskazanej sesji, używaną do połączenia z Redis.
 <SessionName> - określa nazwę sesji, która ma być używana w kluczach pozycji.

Plugins.Redis.Timeout

Ustaw limit czasu wykonania żądania (czas w sekundach, przez jaki należy czekać na zakończenie żądania przed jego przerwaniem).

Domyślnie: `global timeout`
 Zakres: 1–30

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Wtyczka SMART

Omówienie

Parametry konfiguracyjne umożliwiają dostosowanie zachowania wtyczki SMART.

Zaleca się określanie ich we własnym pliku konfiguracyjnym (np. `smart.conf`), a nie w pliku konfiguracyjnym agenta 2. Następnie użyj dyrektywy Include, aby dołączyć ten plik do konfiguracji Zabbix agent 2.

Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest obowiązkowy.

Ta strona jest podzielona na:

- Omówienie parametrów (aby zobaczyć pełne szczegóły parametru, kliknij jego nazwę)
- Szczegóły parametrów

Parameter	Description
Plugins.Smart.Path	Określa ścieżkę do pliku wykonywalnego smartctl.
Plugins.Smart.Timeout	Ustawia limit czasu wykonania żądania (czas w sekundach, przez jaki należy czekać na zakończenie żądania przed jego przerwaniem).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości zawarte w dostarczonych plikach konfiguracyjnych;

- Wartości obsługują zmienne środowiskowe;
- Ścieżka do pliku wykonywalnego `smartctl` musi być ustawiona albo przez dodanie jego katalogu do systemowej zmiennej środowiskowej `PATH`, albo przez skonfigurowanie `Plugins.Smart.Path`; dotyczy to zarówno systemu Linux, jak i Windows;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od "#" są obsługiwane tylko na początku wiersza.

Szczegóły parametru

Plugins.Smart.Path

Określ ścieżkę do pliku wykonywalnego smartctl.

Domyślnie: `smartctl`

Plugins.Smart.Timeout

Ustaw limit czasu wykonania żądania (czas w sekundach, przez jaki należy czekać na zakończenie żądania przed jego przerwaniem).

Domyślnie: `global timeout<br>` Zakres: 1–30

Zobacz także

- Opis ogólnych parametrów konfiguracji Zabbix agent 2: Zabbix agent 2 (UNIX) / Zabbix agent 2 (Windows)
- Instrukcje konfiguracji wtyczek

Protokół wtyczki Zabbix agent 2

Protokół Zabbix agent 2 opiera się na kodzie, rozmiarze i modelu danych.

Kod

Type	Size	Comments
Byte	4	Typ ładunku, obecnie obsługiwany jest tylko JSON.

Rozmiar

Type	Size	Comments
Byte	4	Rozmiar bieżącego ładunku w bajtach.

Dane ładunku

Type	Size	Comments
Byte	Określone przez pole *Size*	Dane w formacie JSON.

Definicja danych ładunku

Dane wspólne

Te parametry są obecne we wszystkich żądaniach/odpowiedziach:

Name	Type	Comments
id	uint32	Dla żądań - inkrementowany identyfikator używany do powiązania żądań z odpowiedziami. Unikalny w obrębie jednego kierunku żądania (tj. od agent do plugin lub od plugin do agent). Dla odpowiedzi - identyfikator odpowiadającego żądania.
type	uint32	Typ żądania.

Żądanie logu

Żądanie wysyłane przez plugin do zapisania komunikatu logu w pliku logu agenta.

direction	plugin → agent
response	no

Parametry specyficzne dla żądań logu:

Name	Type	Comments
severity	uint32	Poziom ważności komunikatu (poziom logowania).
message	string	Komunikat do zapisania w logu.

Przykład:

```
{"id":0,"type":1,"severity":3,"message":"message"}
```

Żądanie rejestracji

Żądanie wysyłane przez agent podczas fazy uruchamiania agent, aby pobrać udostępnione metryki i zarejestrować plugin.

direction	agent → plugin
response	yes

Parametry specyficzne dla żądań rejestracji:

Name	Type	Comments
version	string	Wersja protokołu <major>.<minor>

Przykład:

```
{"id":1,"type":2,"version":"1.0"}
```

Odpowiedź rejestracji

Odpowiedź pluginu na żądanie rejestracji.

direction	plugin → agent
response	n/a

Parametry specyficzne dla odpowiedzi rejestracji:

Name	Type	Comments
name	string	Nazwa pluginu.
metrics	array of strings (optional)	Metryki z opisami używane w pluginie. Zwraca RegisterMetrics(). Nieobecne, jeśli zwracany jest błąd.
interfaces	uint32 (optional)	Maska bitowa interfejsów obsługiwanych przez plugin. Nieobecne, jeśli zwracany jest błąd.
error	string (optional)	Komunikat błędu zwracany, jeśli plugin nie może zostać uruchomiony. Nieobecny, jeśli zwracane są metryki.

Przykłady:

```
{"id":2,"type":3,"metrics":["external.test", "External exporter Test."], "interfaces": 4}
```

lub

```
{"id":2,"type":3,"error":"error message"}
```

Żądanie startu

Żądanie wykonania funkcji Start interfejsu Runner.

direction	agent → plugin
response	no

Żądanie nie ma określonych parametrów, zawiera tylko parametry danych wspólnych.

Przykład:

```
{"id":3,"type":4}
```

Żądanie zakończenia

Żądanie wysyłane przez agent do zamknięcia wtyczki.

direction	agent → plugin
response	no

Żądanie nie ma określonych parametrów, zawiera tylko parametry common data.

Przykład:

```
{"id":3,"type":5}
```

Żądanie eksportu

Żądanie wykonania funkcji Export interfejsu Exporter.

direction	agent → plugin
response	no

Parametry specyficzne dla żądań eksportu:

Name	Type	Comments
key	string	Klucz pluginu.
parameters	array of strings (optional)	Parametry dla funkcji Export.

Przykład:

```
{"id":4,"type":6,"key":"test.key","parameters":["foo","bar"]}
```

Odpowiedź eksportu

Odpowiedź z funkcji Export interfejsu Exporter.

direction	plugin → agent
response	n/a

Parametry specyficzne dla odpowiedzi eksportu:

Name	Type	Comments
value	string (optional)	Wartość odpowiedzi z funkcji Export. Nieobecna, jeśli zwracany jest błąd.
error	string (optional)	Komunikat błędu, jeśli funkcja Export nie została wykonana pomyślnie. Nieobecny, jeśli zwracana jest wartość.

Przykłady:

```
{"id":5,"type":7,"value":"response"}
```

lub

```
{"id":5,"type":7,"error":"error message"}
```

Konfiguracja żądania

Żądanie wykonania funkcji *Configure* interfejsu *Configurator*.

direction	agent → plugin
response	n/a

Parametry specyficzne dla żądań *Configure*:

Name	Type	Comments
global_options	JSON object	Obiekt JSON zawierający globalne opcje konfiguracji agenta.
private_options	JSON object (optional)	Obiekt JSON zawierający prywatne opcje konfiguracji pluginu, jeśli zostały podane.

Przykład:

```
{"id":6,"type":8,"global_options":{...},"private_options":{...}}
```

Walidacja żądania

Żądanie wykonania funkcji *Validate* interfejsu *Configurator*.

direction	agent → plugin
response	yes

Parametry specyficzne dla żądań *Validate*:

Name	Type	Comments
private_options	JSON object (optional)	Obiekt JSON zawierający prywatne opcje konfiguracji pluginu, jeśli zostały podane.

Przykład:

```
{"id":7,"type":9,"private_options":{...}}
```

Walidacja odpowiedzi

Odpowiedź z funkcji *Validate* interfejsu *Configurator*.

direction	plugin → agent
response	n/a

Parametry specyficzne dla odpowiedzi *Validate*:

Name	Type	Comments
error	string (optional)	Komunikat błędu zwracany, jeśli funkcja Validate nie zostanie wykonana pomyślnie. Nieobecny, jeśli wykonanie zakończyło się powodzeniem.

Przykład:

```
{"id":8,"type":10}
```

lub

```
{"id":8,"type":10,"error":"error message"}
```

Java gateway

Przegląd Zabbix Java gateway można zainstalować z kodu źródłowego lub pakietów.

Natywna obsługa monitorowania aplikacji JMX jest dostępna w postaci demona Zabbix o nazwie „Zabbix Java gateway". Zabbix Java gateway to demon napisany w języku Java. Aby uzyskać wartość określonego licznika JMX na hoście, serwer Zabbix wysyła zapytanie do Zabbix Java gateway, który używa interfejsu API zarządzania JMX do zdalnego wysłania zapytania do wybranej aplikacji. Aplikacja nie wymaga instalowania żadnego dodatkowego oprogramowania — musi zostać uruchomiona z opcją `-Dcom.sun.management.jmxremote` w wierszu poleceń.

Java gateway akceptuje połączenia przychodzące z serwera Zabbix lub proxy i może być używany wyłącznie jako „proxy pasywny". W przeciwieństwie do proxy Zabbix może być również używany z proxy Zabbix (proxy Zabbix nie można łączyć łańcuchowo). Dostęp do każdego Java gateway konfiguruje się bezpośrednio w pliku konfiguracyjnym Zabbix serwera lub Zabbix proxy, dlatego na jeden serwer Zabbix lub proxy Zabbix można skonfigurować tylko jeden Java gateway. Jeśli host będzie zawierał pozycje typu *JMX agent* oraz pozycje innego typu, tylko pozycje *JMX agent* zostaną przekazane do Java gateway w celu pobrania danych.

Gdy pozycja musi zostać zaktualizowana za pośrednictwem Java gateway, serwer Zabbix lub proxy łączy się z Java gateway i żąda wartości, którą Java gateway pobiera, a następnie przekazuje z powrotem do serwera lub proxy. W związku z tym Java gateway nie buforuje żadnych wartości.

Serwer Zabbix lub proxy ma określony typ procesów, które łączą się z Java gateway. Ich liczba jest kontrolowana przez opcję `StartJavaPollers`. Wewnętrznie Java gateway uruchamia wiele wątków, których liczba jest kontrolowana przez opcję `START_POLLERS`. Po stronie serwera, jeśli połączenie trwa dłużej niż liczba sekund określona w parametrze „Timeout", zostanie ono zakończone, ale Java gateway może nadal zajmować się pobieraniem wartości z licznika JMX. Aby rozwiązać ten problem, Java gateway udostępnia opcję `TIMEOUT`, która umożliwia ustawienie limitu czasu dla operacji sieciowych JMX.

Zobacz także:

- Parametry konfiguracji: Zabbix Java gateway
- Konfiguracja z pakietów Debian/Ubuntu
- Konfiguracja z pakietów RHEL
- Konfiguracja ze źródeł

Serwer Zabbix lub proxy będzie próbował możliwie najbardziej grupować żądania do pojedynczego celu JMX (zależnie od interwałów pozycji) i wysyłać je do Java gateway w jednym połączeniu, aby uzyskać lepszą wydajność.

Zaleca się, aby `StartJavaPollers` było mniejsze lub równe `START_POLLERS`; w przeciwnym razie mogą wystąpić sytuacje, w których w Java gateway nie będzie dostępnych żadnych wątków do obsługi przychodzących żądań. W takich przypadkach Java gateway używa `ThreadPoolExecutor.CallerRunsPolicy`, co oznacza, że główny wątek obsłuży przychodzące żądanie i tymczasowo nie będzie przyjmował żadnych nowych żądań.

Jeśli próbujesz monitorować aplikacje Java oparte na Wildfly za pomocą Zabbix Java gateway, zainstaluj najnowszy plik `jboss-client.jar` dostępny na stronie pobierania Wildfly.

Parametry konfiguracji: Zabbix Java gateway

Jeśli używasz skryptów `startup.sh` i `shutdown.sh` do uruchamiania Zabbix Java gateway, możesz określić niezbędne parametry konfiguracji w pliku `settings.sh`. Skrypty uruchamiania i zamykania wczytują plik ustawień i zajmują się konwersją zmiennych powłoki (wymienionych w pierwszej kolumnie) na właściwości Java (wymienione w drugiej kolumnie).

Jeśli uruchamiasz Zabbix Java gateway ręcznie, wywołując bezpośrednio `java`, to odpowiednie właściwości Java określasz w wierszu poleceń.

Variable	Property	Mandatory	Range	Default	Description
LISTEN_IP	zabbix.listenIP	no		0.0.0.0	Adres IP, na którym ma nasłuchiwać.
LISTEN_PORT	zabbix.listenPort	no	1024-32767	10052	Port, na którym ma nasłuchiwać.
PID_FILE	zabbix.pidFile	no		/tmp/zabbix_java.pid	Nazwa pliku PID. Jeśli zostanie pominięta, Zabbix Java Gateway zostanie uruchomiony jako aplikacja konsolowa.

Variable	Property	Mandatory	Range	Default	Description
PROPERTIES_FILE	zabbix.propertiesFile	no			Nazwa pliku właściwości. Może służyć do ustawiania dodatkowych właściwości w formacie klucz-wartość w taki sposób, aby nie były widoczne w wierszu poleceń, lub do nadpisywania istniejących. Na przykład: ”javax.net.ssl.trustStorePassword=<p
START_POLLERS	zabbix.startPollers	no	1-1000	5	Liczba wątków roboczych do uruchomienia.
TIMEOUT	zabbix.timeout	no	1-30	3	Czas oczekiwania na operacje sieciowe (w sekundach).

Warning:
Port 10052 nie jest zarejestrowany przez IANA.

Instalacja z kodu źródłowego

Przegląd

Jeśli Zabbix został zainstalowany z kodu źródłowego, poniższe informacje pomogą w konfiguracji Zabbix Java gateway.

Przegląd plików

Jeśli Java gateway została uzyskana ze źródeł, w katalogu `$PREFIX/sbin/zabbix_java` powinien znajdować się zestaw skryptów powłoki, plików JAR i plików konfiguracyjnych. Poniżej podsumowano rolę tych plików.

`bin/zabbix-java-gateway-$VERSION.jar`

Sam plik JAR 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
```

Zależności Java gateway: biblioteki Logback, SLF4J i Android JSON.

```
lib/logback.xml
lib/logback-console.xml
```

Pliki konfiguracyjne Logback.

```
shutdown.sh
startup.sh
```

Pomocnicze skrypty służące do uruchamiania i zatrzymywania Java gateway.

`settings.sh`

Plik konfiguracyjny wczytywany przez powyższe skrypty uruchamiania i zatrzymywania.

Konfigurowanie i uruchamianie Java gateway

Domyślnie Java gateway nasłuchuje na porcie 10052. Jeśli planujesz uruchomić Java gateway na innym porcie, możesz to określić w skrypcie `settings.sh`. Zobacz opis pliku konfiguracyjnego Java gateway, aby dowiedzieć się, jak określić tę i inne opcje.

Warning:
Port 10052 nie jest zarejestrowany przez IANA.

Gdy ustawienia będą już odpowiednie, możesz uruchomić Java gateway, wykonując skrypt `startup.sh`:

```
./startup.sh
```

Analogicznie, gdy Java gateway nie będzie już potrzebny, uruchom skrypt `shutdown.sh`, aby go zatrzymać:

```
./shutdown.sh
```

Należy pamiętać, że w przeciwieństwie do serwer lub proxy, Java gateway jest lekki i nie wymaga bazy danych.

Konfigurowanie serwera do użycia z Java gateway

Gdy Java gateway jest uruchomiony i działa, musisz wskazać serwerowi Zabbix, gdzie ma znaleźć Zabbix Java gateway. Robi się to przez określenie parametrów `JavaGateway` i `JavaGatewayPort` w pliku konfiguracyjnym serwera. Jeśli host, na którym działa aplikacja JMX, jest monitorowany przez proxy Zabbix, zamiast tego należy określić parametry połączenia w pliku konfiguracyjnym proxy.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

Domyślnie serwer nie uruchamia żadnych procesów związanych z monitorowaniem JMX. Jeśli jednak chcesz z niego korzystać, musisz określić liczbę wcześniej utworzonych instancji Java pollers. Robi się to w taki sam sposób, w jaki określa się zwykłe pollers i trappers.

```
StartJavaPollers=5
```

Nie zapomnij zrestartować serwera lub proxy po zakończeniu ich konfiguracji.

Debugowanie Java gateway

W przypadku problemów z Java gateway lub gdy komunikat o błędzie dotyczący pozycja w frontendzie nie jest wystarczająco opisowy, warto zajrzeć do pliku dziennika Java gateway.

Domyślnie Java gateway zapisuje swoje działania do pliku `/tmp/zabbix_java.log` z poziomem logowania "info". Czasami te informacje nie są wystarczające i potrzebne są dane na poziomie logowania "debug". Aby zwiększyć poziom logowania, zmodyfikuj plik `lib/logback.xml` i zmień atrybut level znacznika `<root>` na debug:

```
<root level="debug">
  <appender-ref ref="FILE" />
</root>
```

Należy pamiętać, że w przeciwieństwie do serwer lub proxy Zabbix, po zmianie pliku `logback.xml` nie ma potrzeby ponownego uruchamiania Java gateway Zabbix - zmiany w `logback.xml` zostaną automatycznie uwzględnione. Po zakończeniu debugowania możesz przywrócić poziom logowania do "info".

Jeśli chcesz zapisywać logi do innego pliku lub całkowicie innego medium, na przykład bazy danych, dostosuj plik `logback.xml` do swoich potrzeb. Więcej informacji znajdziesz w Logback Manual.

Czasami w celach debugowania przydatne jest uruchomienie Java gateway jako aplikacji konsolowej zamiast demona. Aby to zrobić, zakomentuj zmienną `PID_FILE` w `settings.sh`. Jeśli `PID_FILE` zostanie pominięta, skrypt `startup.sh` uruchomi Java gateway jako aplikację konsolową i sprawi, że Logback będzie używać pliku `lib/logback-console.xml`, który nie tylko zapisuje logi do konsoli, ale ma również włączony poziom logowania "debug".

Na koniec należy pamiętać, że ponieważ Java gateway używa SLF4J do logowania, możesz zastąpić Logback wybranym przez siebie frameworkiem, umieszczając odpowiedni plik JAR w katalogu lib. Więcej informacji znajdziesz w SLF4J Manual.

Monitorowanie JMX

Zobacz stronę Monitorowanie JMX, aby uzyskać więcej informacji.

Konfiguracja z pakietów Debian/Ubuntu

Przegląd

Jeśli zainstalowano z pakietów, poniższe informacje pomogą w konfiguracji Zabbix Java gateway.

Konfigurowanie i uruchamianie Java gateway

Konfigurację Java gateway można dostosować w pliku:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Więcej informacji można znaleźć w sekcji konfiguracji Zabbix Java gateway parametry.

Aby uruchomić Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Aby automatycznie uruchamiać Zabbix Java gateway podczas startu systemu:

```
systemctl enable zabbix-java-gateway
```

Konfigurowanie serwera do użycia z Java gateway

Gdy Java gateway jest uruchomiony i działa, musisz poinformować serwer Zabbix, gdzie ma znaleźć Zabbix Java gateway. Robi się to przez określenie parametrów `JavaGateway` i `JavaGatewayPort` w pliku konfiguracyjnym serwera. Jeśli host, na którym działa aplikacja JMX, jest monitorowany przez proxy Zabbix, zamiast tego należy określić parametry połączenia w pliku konfiguracyjnym proxy.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

Domyślnie serwer nie uruchamia żadnych procesów związanych z monitorowaniem JMX. Jeśli jednak chcesz z niego korzystać, musisz określić liczbę wstępnie rozwidlonych instancji pollerów Java. Robi się to w taki sam sposób, w jaki określa się zwykłe pollery i trappers.

```
StartJavaPollers=5
```

Nie zapomnij ponownie uruchomić serwera lub proxy po zakończeniu konfiguracji.

Debugowanie Java gateway

Plik dziennika Zabbix Java gateway to:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Jeśli chcesz zwiększyć poziom logowania, edytuj plik:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

i zmień `info` w `level="info"` na debug lub nawet `trace` (do zaawansowanego rozwiązywania problemów):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
      <root level="info">
              <appender-ref ref="FILE" />
      </root>

</configuration>
```

Monitorowanie JMX

Zobacz stronę Monitorowanie JMX, aby uzyskać więcej informacji.

Konfiguracja z pakietów RHEL

Przegląd

Jeśli zainstalowano z pakietów, poniższe informacje pomogą w konfiguracji Zabbix Java gateway.

Konfigurowanie i uruchamianie Java gateway

Parametry konfiguracji Zabbix Java gateway można dostosować w pliku:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Więcej informacji można znaleźć w parametrach konfiguracji Zabbix Java gateway.

Aby uruchomić Zabbix Java gateway:

```
systemctl restart zabbix-java-gateway
```

Aby automatycznie uruchamiać Zabbix Java gateway podczas startu systemu:

RHEL 7 i nowsze:

```
systemctl enable zabbix-java-gateway
```

RHEL starsze niż 7:

```
chkconfig --level 12345 zabbix-java-gateway on
```

Konfigurowanie serwera do użycia z Java gateway

Gdy Java gateway jest uruchomiony i działa poprawnie, należy wskazać serwerowi Zabbix, gdzie ma znaleźć Zabbix Java gateway. W tym celu należy określić parametry `JavaGateway` i `JavaGatewayPort` w pliku konfiguracyjnym serwera. Jeśli host, na którym działa aplikacja JMX, jest monitorowany przez proxy Zabbix, parametry połączenia należy zamiast tego określić w pliku konfiguracyjnym proxy.

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

Domyślnie serwer nie uruchamia żadnych procesów związanych z monitorowaniem JMX. Jeśli jednak chcesz z niego korzystać, musisz określić liczbę wstępnie rozwidlonych instancji Java pollers. Robi się to w taki sam sposób, jak określa się zwykłe pollers i trappers.

```
StartJavaPollers=5
```

Nie zapomnij ponownie uruchomić serwera lub proxy po zakończeniu konfiguracji.

Debugowanie Java gateway

Plik dziennika Java gateway Zabbix to:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Jeśli chcesz zwiększyć poziom rejestrowania, edytuj plik:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

i zmień `info` w `level="info"` na debug lub nawet `trace` (w przypadku zaawansowanego rozwiązywania problemów):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
      <root level="info">
              <appender-ref ref="FILE" />
      </root>

</configuration>
```

Monitorowanie JMX

Zobacz stronę Monitorowanie JMX, aby uzyskać więcej informacji.

Nadawca

Przegląd Zabbix sender to narzędzie wiersza poleceń, którego można używać do wysyłania danych dotyczących wydajności do serwera Zabbix lub proxy w celu ich przetwarzania.

Narzędzie jest zwykle używane w długo działających skryptach użytkownika do okresowego wysyłania danych dotyczących dostępności i wydajności.

Aby wysyłać wyniki bezpośrednio do serwera Zabbix lub proxy, należy skonfigurować pozycję typu *Zabbix trapper*.

Zobacz także:

- Protokół Zabbix sender
- Biblioteka dynamiczna Zabbix sender dla systemu Windows
- Biblioteka Python dla Zabbix, która ma wbudowaną funkcjonalność umożliwiającą działanie jak Zabbix sender

Uruchamianie Zabbix sender Przykład uruchomienia Zabbix sender w systemie UNIX:

```
cd bin
./zabbix_sender -z zabbix -s "Linux DB3" -k db.connections -o 43
```

gdzie:

- `z` - host serwera Zabbix (można również użyć adresu IP);
- `s` - nazwa techniczna monitorowanego hosta (zarejestrowana w frontendzie Zabbix);
- `k` - klucz pozycji;
- `o` - wartość do wysłania.

Attention:
Opcje zawierające białe znaki muszą być ujęte w cudzysłowy.

Zabbix sender może służyć do wysyłania wielu wartości z pliku wejściowego. Więcej informacji można znaleźć na stronie podręcznika Zabbix sender.

Jeśli określono plik konfiguracyjny, Zabbix sender używa wszystkich adresów zdefiniowanych w parametrze konfiguracyjnym agenta `ServerActive` do wysyłania danych. Jeśli wysyłanie do jednego adresu nie powiedzie się, sender próbuje wysłać dane do pozostałych adresów. Jeśli wysyłanie danych wsadowych do jednego adresu nie powiedzie się, kolejne partie danych nie są wysyłane na ten adres.

Zabbix sender akceptuje ciągi znaków zakodowane w UTF-8 (zarówno w systemach uniksopodobnych, jak i w systemie Windows), bez znacznika kolejności bajtów (BOM) na początku pliku.

Zabbix sender w systemie Windows można uruchomić w podobny sposób:

```
zabbix_sender.exe [options]
```

Scenariusze wysyłania w czasie rzeczywistym za pomocą `zabbix_sender` gromadzą wiele przekazanych do niego wartości, które pojawiają się w krótkich odstępach czasu, i wysyłają je do serwera w ramach jednego połączenia. Wartość, której odstęp od poprzedniej wartości nie przekracza 0,2 sekundy, może zostać umieszczona w tym samym stosie, jednak maksymalny czas odpytywania nadal wynosi 1 sekundę.

Note:
Zabbix sender zakończy działanie, jeśli w określonym pliku konfiguracyjnym znajduje się nieprawidłowy wpis parametru (niezgodny z notacją `parameter=value`).

Uruchamianie Zabbix sender z wykrywaniem niskiego poziomu Przykład uruchomienia Zabbix sender w celu wysłania wartości w formacie JSON dla wykrywania niskiego poziomu:

```
./zabbix_sender -z 192.168.1.113 -s "Zabbix serwer" -k trapper.discovery.item -o '[{"{#FSNAME}":"/","{#FST
```

Aby to działało, reguła wykrywania niskiego poziomu musi mieć typ pozycja *Zabbix trapper* (w tym przykładzie z kluczem `trapper.discovery.item`).

Protokół Zabbix sender

Serwer Zabbix i proxy Zabbix używają protokołu komunikacji opartego na JSON do odbierania danych od Zabbix sender. Dane mogą być odbierane za pomocą pozycji trapper lub pozycji HTTP agent z włączonym trappingiem.

Wiadomości żądania i odpowiedzi muszą zaczynać się od nagłówka i długości danych.

Żądanie Zabbix sender

```
{
    "request": "sender data",
    "data": [
        {
            "host": "<hostname>",
            "key": "trap",
            "value": "test value"
        }
    ]
}
```

Odpowiedź serwera Zabbix

```
{
    "response": "success",
    "info": "processed: 1; failed: 0; total: 1; seconds spent: 0.060753"
}
```

Żądanie Zabbix sender z sygnaturą czasową

Alternatywnie, Zabbix sender może wysłać żądanie z sygnaturą czasową i nanosekundami.

```
{
    "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
}
```

Odpowiedź serwera Zabbix

```
{
    "response": "success",
    "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.060904"
}
```

Biblioteka dynamiczna Zabbix sender dla Windows

W środowisku Windows aplikacje mogą wysyłać dane do serwera/proxy Zabbix za pomocą biblioteki dynamicznej Zabbix sender (`zabbix_sender.dll`) zamiast uruchamiać zewnętrzny proces (`zabbix_sender.exe`).

Pliki zabbix_sender.h i zabbix_sender.lib są wymagane do kompilowania aplikacji użytkownika z użyciem zabbix_sender.dll.

Pobieranie

Istnieją dwa sposoby uzyskania zabbix_sender.dll.

1. Pobierz pliki zabbix_sender.h, zabbix_sender.lib i zabbix_sender.dll jako archiwum ZIP.

Podczas wybierania opcji pobierania upewnij się, że w sekcji *Encryption* wybrano opcję "No encryption", a w sekcji *Packaging* opcję "Archive". Następnie pobierz Zabbix agent (nie Zabbix agent 2).

Pliki zabbix_sender.h, zabbix_sender.lib i zabbix_sender.dll będą znajdować się w pobranym archiwum ZIP w katalogu `bin\dev`. Rozpakuj pliki w wybranej lokalizacji.

2. Zbuduj zabbix_sender.dll ze źródeł (zobacz instrukcje).

Biblioteka dynamiczna wraz z plikami deweloperskimi będzie znajdować się w katalogu `bin\winXX\dev`. Aby z niej korzystać, dołącz plik nagłówkowy zabbix_sender.h i połącz aplikację z biblioteką zabbix_sender.lib.

Zobacz także

- przykład prostego narzędzia Zabbix sender zaimplementowanego z użyciem biblioteki dynamicznej Zabbix sender, aby zilustrować sposób korzystania z biblioteki;
- plik zabbix_sender.h zawierający funkcje interfejsu biblioteki dynamicznej Zabbix sender. Plik ten zawiera dokumentację wyjaśniającą przeznaczenie każdej funkcji interfejsu, jej argumenty oraz wartość zwracaną.

Pobierz

Przegląd Zabbix get to narzędzie wiersza poleceń, które można używać do komunikacji z agent i pobierania wymaganych informacji z agenta.

To narzędzie jest zwykle używane do rozwiązywania problemów z agentami Zabbix.

Zobacz także bibliotekę Python dla Zabbix, która ma wbudowaną funkcjonalność pozwalającą działać podobnie jak Zabbix get.

Uruchamianie Zabbix get Przykład uruchomienia Zabbix get w systemie UNIX w celu pobrania wartości obciążenia procesora z agenta:

```
cd bin
./zabbix_get -s 127.0.0.1 -p 10050 -k system.cpu.load[all,avg1]
```

Inny przykład uruchomienia Zabbix get w celu pobrania ciągu znaków ze strony internetowej:

```
cd bin
./zabbix_get -s 192.168.1.1 -p 10050 -k "web.page.regexp[www.example.com,,,\"USA: ([a-zA-Z0-9.-]+)\",,\1]"
```

Zwróć uwagę, że klucz pozycji zawiera tutaj spację, więc do oznaczenia klucza pozycji dla powłoki użyto cudzysłowów. Cudzysłowy nie są częścią klucza pozycji; zostaną usunięte przez powłokę i nie zostaną przekazane do agenta Zabbix.

Jeśli klucz pozycji nie jest obsługiwany, Zabbix get zwróci kod wyjścia 1.

Zabbix get akceptuje następujące parametry wiersza poleceń:

```
-s --host <host name or IP>             Określa nazwę hosta lub adres IP hosta
-p --port <port number>                 Określa numer portu agenta działającego na hoście (domyślnie: 1005
-I --source-address <IP address>        Określa źródłowy adres IP
-t --timeout <seconds>                  Określa limit czasu. Prawidłowy zakres: 1-600 sekund (domyślnie: 3
-k --key <item key>                     Określa klucz pozycji, dla której ma zostać pobrana wartość
-P --protocol <value>                   Protokół używany do komunikacji z agentem. Wartości:
                                            auto - łączy przy użyciu protokołu JSON, w razie niepowodzenia
                                            json - łączy przy użyciu protokołu JSON
                                            plaintext - łączy przy użyciu protokołu tekstowego, w którym w
-h --help                               Wyświetla ten komunikat pomocy
-V --version                            Wyświetla numer wersji

--tls-connect <value>                   Sposób łączenia się z agentem. Wartości:
                                            unencrypted - łączy bez szyfrowania (domyślnie)
                                            psk - łączy przy użyciu TLS i współdzielonego klucza
                                            cert - łączy przy użyciu TLS i certyfikatu
--tls-ca-file <CA file>                 Pełna ścieżka do pliku zawierającego certyfikaty głównego urzędu C
--tls-crl-file <CRL file>               Pełna ścieżka do pliku zawierającego certyfikaty unieważnione
--tls-agent-cert-issuer <cert issuer>   Dozwolony wystawca certyfikatu agenta
--tls-agent-cert-subject <cert subject> Dozwolony podmiot certyfikatu agenta
--tls-cert-file <cert file>             Pełna ścieżka do pliku zawierającego certyfikat lub łańcuch certyf
--tls-key-file <key file>               Pełna ścieżka do pliku zawierającego klucz prywatny
--tls-psk-identity <PSK-identity>       Unikalny, rozróżniający wielkość liter ciąg używany do identyfikac
--tls-psk-file <PSK-file>               Pełna ścieżka do pliku zawierającego współdzielony klucz
--tls-cipher13 <cipher-string>          Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego dla TLS 1.3. Zastępuje
--tls-cipher <cipher-string>            Ciąg priorytetów GnuTLS (dla TLS 1.2 i nowszych) lub ciąg szyfrów
```

Więcej informacji można znaleźć w stronie podręcznika Zabbix get.

Zabbix get w systemie Windows można uruchomić w podobny sposób:

```
zabbix_get.exe [options]
```

JS

Przegląd `zabbix_js` to narzędzie wiersza poleceń, które można używać do testowania skryptów osadzonych.

To narzędzie wykona skrypt użytkownika z parametrem typu string i wyświetli wynik. Skrypty są wykonywane przy użyciu osadzonego silnika skryptowego Zabbix.

W przypadku błędów kompilacji lub wykonania `zabbix_js` wyświetli błąd na stderr i zakończy działanie z kodem 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
```

Użycie zabbix_js akceptuje następujące parametry wiersza poleceń:

```
-s, --script script-file          Określa nazwę pliku skryptu do wykonania
                                  Jeśli jako nazwę pliku zostanie podane "-", skrypt zostanie odczytany ze
-i, --input input-file            Określa nazwę pliku z informacjami wejściowymi
                                  Zawartość zostanie przekazana do skryptu jako wartość
                                  Wzajemnie wykluczające się z -p
                                  Jeśli jako nazwę pliku zostanie podane "-", skrypt zostanie odczytany ze
-p, --param input-param           Określa parametr wejściowy: zmienną, która zostanie przekazana do skrypt
                                  Wzajemnie wykluczające się z -i
-l, --loglevel log-level          Określa poziom logowania
                                  Zakres: 0-5
-t, --timeout timeout             Określa limit czasu w sekundach
                                  Dozwolony zakres: 1-600 sekund (domyślnie: 10 sekund)
-h, --help                        Wyświetla informacje pomocy
-V, --version                     Wyświetla numer wersji
-w <webdriver url>                Włącza monitorowanie przeglądarki
```

Przykłady: Przykład 1: Wykonanie skryptu ze zmienną zawierającą wartość

Plik ze skryptem (script-file.js):

```
return value;
```

Użycie:

```
zabbix_js -s script-file.js -p example_value
```

Wyjście: example_value

Przykład 2: Wykonanie skryptu z plikiem zawierającym informacje wejściowe

Plik ze skryptem (script-file.js):

```
return value;
```

Plik z informacjami wejściowymi (example.txt):

```
Przykład informacji wejściowych z pliku
```

Użycie:

```
zabbix_js -s script-file.js -i example.txt
```

Wyjście: Przykład informacji wejściowych z pliku

Przykład 3: Wykonanie skryptu z danymi wejściowymi odczytanymi ze stdin

Plik ze skryptem (script-file.js):

```
return value;
```

Użycie:

```
zabbix_js -s script-file.js -i -
```

Dane wejściowe (stdin):

```
Przykład danych wejściowych ze stdin
```

Wyjście: Przykład danych wejściowych ze stdin

Usługa sieciowa

Omówienie Usługa webowa Zabbix to proces używany do komunikacji z zewnętrznymi usługami webowymi. Obecnie usługa webowa Zabbix jest używana do generowania i wysyłania zaplanowanych raportów, a w przyszłości planowane jest dodanie dodatkowych funkcji.

Serwer Zabbix łączy się z usługą webową za pośrednictwem HTTP(S). Usługa webowa Zabbix wymaga zainstalowania Google Chrome na tym samym hoście; w niektórych dystrybucjach usługa może również działać z Chromium (zobacz znane problemy).

Instalacja Oficjalny pakiet zabbix-web-service jest dostępny w repozytorium Zabbix.

Aby skompilować usługę internetową Zabbix ze źródeł, określ opcję konfiguracji `--enable-webservice`.

Aby skonfigurować usługę internetową Zabbix, zaktualizuj parametry pliku konfiguracyjnego usługi internetowej.

Attention:
Zdecydowanie zaleca się skonfigurowanie szyfrowania między serwerem Zabbix a usługą internetową Zabbix przy użyciu certyfikatów. Domyślnie dane przesyłane między serwerem Zabbix a usługą internetową Zabbix nie są szyfrowane, co może prowadzić do nieautoryzowanego dostępu.

Parametry konfiguracji: Zabbix web service

Omówienie

Usługa webowa Zabbixa to proces używany do komunikacji z zewnętrznymi usługami webowymi.

Parametry obsługiwane przez plik konfiguracyjny usługi webowej Zabbixa (`zabbix_web_service.conf`) są wymienione w tej sekcji.

Parametry są podane bez dodatkowych informacji. Kliknij parametr, aby zobaczyć pełne szczegóły.

Parameter	Description
AllowedIP	Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbixa i proxy Zabbixa.
DebugLevel	Poziom debugowania.
IgnoreURLCertErrors	Określa sposób obsługi błędów weryfikacji certyfikatu TLS podczas uzyskiwania dostępu do URL frontend.
Include	Możesz dołączyć pojedyncze pliki lub wszystkie pliki w katalogu do pliku konfiguracyjnego.
ListenPort	Usługa będzie nasłuchiwać na tym porcie na połączenia z serwera.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogType	Typ wyjścia dziennika.
Timeout	Maksymalny czas (w sekundach) poświęcony na formatowanie raportu PDF report pulpitu.
TLSAccept	Jakie połączenia przychodzące akceptować.
TLSCAFile	Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu peer, używane do szyfrowanej komunikacji między komponentami Zabbixa.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat usługi lub łańcuch certyfikatów, używane do szyfrowanej komunikacji między komponentami Zabbixa.
TLSKeyFile	Pełna ścieżka do pliku zawierającego prywatny klucz usługi, używane do szyfrowanej komunikacji między komponentami Zabbixa.

Wszystkie parametry są opcjonalne, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Należy pamiętać, że:

- wartości domyślne odzwierciedlają domyślne ustawienia procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- wartości obsługują zmienne środowiskowe;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametru

AllowedIP

Lista adresów IP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS serwerów Zabbix i proxy Zabbix. Połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj.
Jeśli obsługa IPv6 jest włączona, to `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` pozwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` można użyć, aby zezwolić na dowolny adres IPv4.

Przykład:

```
127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
```

Wymagane: tak

DebugLevel

Określ poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - do debugowania (generuje dużo informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Domyślnie: 3
 Zakres: 0-5

IgnoreURLCertErrors

Określa sposób obsługi błędów weryfikacji certyfikatu TLS podczas uzyskiwania dostępu do URL frontend:
0 - nie ignoruj błędów certyfikatu;
1 - ignoruj błędy certyfikatu.

Domyślnie: 0
 Zakres: 0-1

Include

Określ katalog lub pojedyncze pliki, które mają zostać dołączone do pliku konfiguracyjnego.
 Podczas instalacji Zabbix ten parametr jest ustawiany na `/usr/local/etc`, chyba że zmienisz to podczas kompilacji.

Ogólne zasady dla parametru `Include`:

- Ścieżka może być względna względem lokalizacji pliku `zabbix_web_service.conf`.
- W jednym pliku konfiguracyjnym można określić wiele parametrów `Include`.
- Zabbix przetwarza parametr `Include` natychmiast po jego odczytaniu, więc warto umieścić go na końcu pliku konfiguracyjnego.

Zasady dotyczące dołączanych plików:

- Każdy plik musi być czytelny.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie uruchomi się.

Zasady dotyczące dołączania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności dołączania (np. pliki nie są dołączane alfabetycznie).
- Nie definiuj tego samego parametru w więcej niż jednym dołączonym pliku (np. aby nadpisać ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Zwracaj uwagę na kopie zapasowe, które niektóre edytory tekstu tworzą automatycznie (np. edycja `include/my_specific.conf` może utworzyć `include/my_specific.conf.BAK`), ponieważ oba pliki zostaną dołączone.

Aby dołączyć tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (`*`) do dopasowywania wzorca.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenPort

Usługa będzie nasłuchiwać na tym porcie na połączenia z serwera.

Domyślnie: 10053
 Zakres: 1024-32767

LogFile

Nazwa pliku dziennika.

Przykład:

```
/tmp/zabbix_web_service.log
```

Wymagane: Tak, jeśli `LogType` jest ustawione na `file`; w przeciwnym razie nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację dziennika.
Uwaga: Jeśli zostanie osiągnięty limit rozmiaru pliku dziennika i rotacja pliku nie powiedzie się z jakiegokolwiek powodu, istniejący plik dziennika zostanie obcięty i utworzony od nowa.

Domyślnie: 1
 Zakres: 0-1024

LogType

Typ wyjścia logu: - *file* - zapisuj log do pliku określonego parametrem LogFile; - *system* - zapisuj log do sysloga; - *console* - zapisuj log do standardowego wyjścia.

Domyślnie: `file`

Limit czasu

Maksymalny czas (w sekundach) poświęcony na formatowanie raportu PDF dashboardu.

Domyślnie: 10
 Zakres: 1-30

TLSAccept

Jakie połączenia przychodzące akceptować: - *unencrypted* - akceptuj połączenia bez szyfrowania (domyślnie); - *cert* - akceptuj połączenia z TLS i certyfikatem.

Domyślnie: `unencrypted`

TLSCAFile

Pełna ścieżka do pliku zawierającego certyfikaty głównych urzędów CA do weryfikacji certyfikatu partnera, używana do szyfrowanej komunikacji między komponentami Zabbix.

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat usługi lub łańcuch certyfikatów, używany do szyfrowanej komunikacji z komponentami Zabbix.

TLSKeyFile

Pełna ścieżka do pliku zawierającego prywatny klucz usługi, używany do szyfrowanej komunikacji między komponentami Zabbix.

Adapter mostu

Adapter mostu Zabbix umożliwia komunikację serwera Zabbix z aplikacją mobilną Zabbix.

Aplikacja mobilna Zabbix nie łączy się bezpośrednio z serwerem Zabbix. Zamiast tego dwa komponenty przekazują między sobą dane:

- Zabbix Bridge - centralnie hostowana usługa, która wymienia dane z aplikacją mobilną Zabbix na urządzeniu mobilnym.
- Adapter mostu Zabbix - komponent instalowany i uruchamiany przez użytkownika. Łączy serwer Zabbix z usługą Zabbix Bridge.

Adapter mostu można zainstalować w tym samym systemie co serwer Zabbix lub w innym systemie, korzystając z pakietów.

Aby skonfigurować adapter mostu Zabbix, zaktualizuj jego plik konfiguracyjny (domyślna ścieżka: `/etc/zabbix/zabbix_bridge_adapter`

Aby zapewnić bezpieczeństwo adaptera mostu Zabbix, zobacz Bezpieczny dostęp mobilny.

Przepływ danych Podczas korzystania z aplikacji mobilnej Zabbix dane przepływają w następujący sposób:

1. Wykonujesz działanie w aplikacji mobilnej Zabbix, która następnie wysyła zapytanie do Zabbix Bridge.
2. Zabbix Bridge przekazuje zapytanie do adaptera Zabbix bridge jako żądanie JSON-RPC.
3. Adapter Zabbix bridge sprawdza, czy żądana metoda Zabbix API jest dozwolona.
   Jeśli metoda nie jest dozwolona, adapter bridge odrzuca żądanie.
   Jeśli metoda jest dozwolona, adapter bridge wysyła żądanie do Zabbix API.
4. Zabbix API przetwarza żądanie i zwraca odpowiedź do adaptera Zabbix bridge.
5. Adapter Zabbix bridge zwraca odpowiedź do Zabbix Bridge.
6. Zabbix Bridge przekazuje odpowiedź do aplikacji mobilnej Zabbix, która wyświetla dane na urządzeniu mobilnym.

Parametry konfiguracji: adapter mostu Zabbix

Przegląd

Parametry konfiguracji umożliwiają dostosowanie działania adaptera Zabbix bridge.

Określ wartości parametrów w pliku konfiguracyjnym (`zabbix_bridge_adapter.conf`). Wszystkie parametry są **opcjonalne**, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
AllowedAPIMethods	Określa, które metody API Zabbix mogą być wywoływane za pośrednictwem adaptera Zabbix bridge.
DataDir	Określa katalog, w którym tworzony i przechowywany jest plik stanu adaptera bridge oraz klucze szyfrowania.
DebugLevel	Ustawia poziom debugowania.
Include	Określa katalog lub pojedyncze pliki, które mają zostać uwzględnione w pliku konfiguracyjnym.
ListenIP	Określa adres IP, na którym adapter bridge nasłuchuje przychodzących połączeń HTTP(S) z serwera Zabbix.
ListenHealthPort	Określa port, na którym adapter bridge nasłuchuje przychodzących połączeń HTTP używanych do sprawdzania stanu.
ListenPort	Określa port, na którym adapter bridge nasłuchuje przychodzących połączeń HTTP(S) z serwera Zabbix.
LogFile	Określa pełną ścieżkę do pliku dziennika adaptera bridge.
LogFileSize	Ustawia maksymalny rozmiar pliku dziennika (w MB).
LogType	Określa typ danych wyjściowych dziennika.
PidFile	Określa plik PID.
TLSConfig.AllowedCN	Określa rozdzielaną przecinkami listę dozwolonych nazw Common Name klientów (tylko `mTLS`).
TLSConfig.CAFile	Określa pełną ścieżkę do pliku zawierającego najwyższy certyfikat CA lub certyfikaty CA używane do weryfikowania certyfikatów klientów w trybie `mTLS`.
TLSConfig.CertFile	Określa pełną ścieżkę do pliku zawierającego certyfikat lub łańcuch certyfikatów.
TLSConfig.KeyFile	Określa pełną ścieżkę do pliku zawierającego klucz prywatny.
TLSConfig.Mode	Ustawia tryb TLS dla przychodzących połączeń na porcie `ListenPort`.
ServerId	Określa unikatowy identyfikator serwera Zabbix. **Wymagane**.
ZabbixAPITLSCAFile	Określa pełną nazwę ścieżki do pliku zawierającego najwyższy certyfikat CA lub certyfikaty CA używane przez adapter bridge do weryfikowania certyfikatu API Zabbix.
ZabbixAPITLSConnect	Ustawia tryb TLS dla wychodzących połączeń z API Zabbix.
ZabbixAPIURL	Określa URL punktu końcowego API Zabbix. **Wymagane**.
ZabbixBridgeURL	Określa URL Zabbix Bridge używany do pobierania konfiguracji Zabbix Bridge (pobieranej podczas uruchamiania adaptera bridge).

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne procesu, a nie wartości w dostarczonych plikach konfiguracyjnych.
- Wartości obsługują zmienne środowiskowe.
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8, bez BOM;
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

AllowedAPIMethods

Określ, które metody API Zabbix mogą być wywoływane za pośrednictwem adaptera Zabbix bridge.
 W przypadku kilku metod powtórz parametr, podając jedną metodę w każdym wierszu.

Każda wartość parametru musi być albo dokładną nazwą metody (np. `problem.get`), albo wzorcem wieloznacznym, który zastępuje tylko całą nazwę API (np. `*.get`) lub całą nazwę metody (np. `problem.*`).

Każda inna wartość jest nieprawidłowa i uniemożliwia uruchomienie adaptera Zabbix bridge. Dotyczy to również pojedynczego symbolu wieloznacznego (`*`), częściowego wzorca wieloznacznego (`host*` lub `host*.*`) oraz pustej wartości (`AllowedAPIMethods=`).

Przykład:

```
AllowedAPIMethods=apiinfo.version
AllowedAPIMethods=problem.*
AllowedAPIMethods=*.get
```

Minimalny zestaw metod wymaganych przez aplikację mobilną 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
```

Domyślnie: nieustawione (wszystkie metody API Zabbix są odrzucane)

DataDir

Określ katalog, w którym tworzony i przechowywany jest plik stanu adaptera bridge oraz klucze szyfrowania.
 Katalog musi mieć uprawnienia 0700, a wszystkie przechowywane w nim pliki muszą mieć uprawnienia 0600.

Wartość domyślna: `/var/lib/zabbix-bridge-adapter`

DebugLevel

Ustaw poziom debugowania:

- 0 - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix
- 1 - informacje krytyczne
- 2 - informacje o błędach
- 3 - ostrzeżenia
- 4 - debugowanie (generuje dużo informacji)
- 5 - rozszerzone debugowanie (generuje jeszcze więcej informacji)

Wartość domyślna: 3
 Zakres: 0-5

Include

Określ katalog lub pojedyncze pliki, które mają zostać uwzględnione w pliku konfiguracyjnym.
 Podczas instalacji Zabbix ten parametr jest ustawiany na `/usr/local/etc`, chyba że zmienisz go podczas kompilacji.

Ogólne zasady dotyczące parametru `Include`:

- Ścieżka może być względna względem lokalizacji pliku `zabbix_bridge_adapter.conf`.
- W jednym pliku konfiguracyjnym można określić wiele parametrów `Include`.
- Zabbix przetwarza parametr `Include` natychmiast po jego odczytaniu, dlatego rozważ umieszczenie go na końcu pliku konfiguracyjnego.

Zasady dotyczące uwzględnianych plików:

- Każdy plik musi być możliwy do odczytu.
- Każdy plik musi mieć poprawną składnię. W przeciwnym razie komponent nie zostanie uruchomiony.

Zasady dotyczące uwzględniania katalogów lub wzorców plików:

- Nie zakładaj żadnej kolejności uwzględniania (np. pliki nie są uwzględniane w kolejności alfabetycznej).
- Nie definiuj tego samego parametru w więcej niż jednym uwzględnianym pliku (np. aby zastąpić ustawienie ogólne ustawieniem szczegółowym), ponieważ nie można przewidzieć, która wartość zostanie użyta.
- Zwracaj uwagę na kopie zapasowe tworzone automatycznie przez niektóre edytory tekstu (np. edycja `include/my_specific.conf` może utworzyć `include/my_specific.conf.BAK`), ponieważ oba pliki zostaną uwzględnione.

Aby uwzględnić tylko odpowiednie pliki z katalogu, użyj symbolu wieloznacznego (`*`) do dopasowywania wzorców.

Przykład:

```
Include=/absolute/path/to/config/files/*.conf
```

ListenIP

Określ adres IP nasłuchiwania adaptera bridge dla przychodzących połączeń HTTP(S) z serwera Zabbix.
 Jeśli nie określono, adapter bridge nasłuchuje na wszystkich adresach IP.

ListenHealthPort

Określ port nasłuchiwania adaptera bridge dla przychodzących połączeń HTTP używanych do sprawdzania stanu.
 Udostępnia endpointy `/health` i `/ready`.
 Jeśli nie zostanie określony, usługa sprawdzania stanu nie zostanie uruchomiona.

Wartość domyślna: 8081
 Zakres: 1024-32767

ListenPort

Określ port nasłuchiwania adaptera bridge dla przychodzących połączeń HTTP(S) z serwera Zabbix.

Wartość domyślna: 10054
 Zakres: 1024-32767

LogFile

Określ pełną ścieżkę do pliku dziennika adaptera bridge.

Wymagane: **Tak**, jeśli `LogType` jest ustawione na `file`

Wartość domyślna: `/tmp/zabbix_bridge_adapter.log`

LogFileSize

Ustaw maksymalny rozmiar pliku dziennika (w MB).
 0 - wyłącza automatyczną rotację dziennika.

Wartość domyślna: 1
 Zakres: 0-1024

LogType

Określ typ wyjścia dziennika:

- `file` - zapisuje dziennik do pliku określonego parametrem `LogFile`.
- `system` - zapisuje dziennik do syslog.
- `console` - zapisuje dziennik na standardowe wyjście.

Wartość domyślna: `file`

PidFile

Określ plik PID.

Wartość domyślna: `/tmp/zabbix_bridge_adapter.pid`

ServerId

Określ unikatowy identyfikator serwera Zabbix.
 Dostępny w sekcji *Raporty* > *Informacje o systemie* w interfejsie użytkownika Zabbix.

Wymagane: **Tak**

TLSConfig.AllowedCN

Określ rozdzielaną przecinkami listę dozwolonych nazw Common Name klientów (tylko `mTLS`).
 Jeśli nie określono, adapter bridge nie sprawdza nazwy Common Name klienta.

TLSConfig.CAFile

Określ pełną ścieżkę do pliku zawierającego certyfikat(y) głównego urzędu certyfikacji (CA) używane do weryfikacji certyfikatów klienta w trybie `mTLS`.

TLSConfig.CertFile

Określ pełną ścieżkę do pliku zawierającego certyfikat lub łańcuch certyfikatów.

TLSConfig.KeyFile

Określ pełną ścieżkę do pliku zawierającego klucz prywatny.

TLSConfig.Mode

Ustaw tryb TLS dla połączeń przychodzących na `ListenPort`:

- `none` - akceptuj połączenia bez szyfrowania.
- `tls` - standardowy TLS z minimalną wersją v1.2.
- `mtls` - rozszerza TLS o wzajemne uwierzytelnianie.

Wartość domyślna: `none`

ZabbixAPITLSCAFile

Określ pełną ścieżkę do pliku zawierającego certyfikat(y) głównego urzędu certyfikacji (CA) używane przez adapter bridge do weryfikowania certyfikatu API Zabbix.

Wymagane: **Tak**, jeśli `ZabbixAPITLSConnect` jest ustawione na `verify_ca` lub `verify_full`.

ZabbixAPITLSConnect

Ustaw tryb TLS dla połączeń wychodzących do API Zabbix:

- `disabled` (lub pusta wartość) - używaj zwykłego protokołu HTTP; `ZabbixAPIURL` musi używać `http://`.
- `required` - używaj HTTPS bez weryfikacji certyfikatu (tylko do testów).
- `verify_ca` - używaj HTTPS i weryfikuj certyfikat API Zabbix.
- `verify_full` - używaj HTTPS oraz weryfikuj certyfikat API Zabbix i nazwę hosta.

ZabbixAPIURL

Określ adres URL punktu końcowego API Zabbix.

Wymagane: **Tak**

Przykład:

`ZabbixAPIURL=https://example.com/zabbix/api_jsonrpc.php`

ZabbixBridgeURL

Określ adres URL mostu Zabbix do pobierania konfiguracji Zabbix Bridge (pobieranej podczas uruchamiania adaptera mostu).

Wartość domyślna: `ZabbixBridgeURL=https://test.aws.zabbix-lab.win/.well-known/bridge-configuration`

Serwer MCP

Przegląd serwer MCP Zabbix to proces umożliwiający agentom AI pracę z Zabbix za pomocą Model Context Protocol (MCP). serwer MCP komunikuje się z klientami MCP i API Zabbix.

Za pomocą serwera MCP agent AI może:

- Pobierać grupy hostów, grupy szablonów, hosty, szablony, pozycje, reguły LLD, interfejsy, makra, wyzwalacze, problemy, okresy utrzymania oraz dane historii pozycji (z wyjątkiem pozycji binarnych);
- Wykonywać aktualizacje problemów: dodawać komentarze, potwierdzać lub anulować potwierdzenie, zmieniać ważność, wyciszać lub anulować wyciszenie oraz zamykać problemy;
- Tworzyć, aktualizować i usuwać okresy utrzymania.

Tokeny API Zabbix są używane do autoryzacji; tokeny są wysyłane z każdym żądaniem. Dostęp jest ograniczony uprawnieniami właściciela tokenu: widoczne są tylko dane z grup hostów dostępnych dla tego użytkownika, a działania, do których użytkownik nie ma uprawnień, są odrzucane.

serwer MCP jest bezstanowy: między żądaniami nie jest przechowywana żadna sesja.

> **Note:**
> Chociaż serwer MCP Zabbix może działać ze starszymi lub nowszymi wersjami Zabbix, jego zgodność jest oficjalnie obsługiwana wyłącznie z tą samą wersją Zabbix, co wersja samego serwera MCP.

Zobacz także:

- Parametry konfiguracji: serwer MCP
- Narzędzia i zasoby

Instalacja Oficjalny pakiet zabbix-mcp-server jest dostępny w repozytorium Zabbix.

Aby zainstalować Zabbix MCP server ze źródeł, zbuduj go za pomocą polecenia `make`.

Aby skonfigurować Zabbix MCP server, zaktualizuj parametry pliku konfiguracyjnego MCP server. Należy ustawić co najmniej parametry `FrontendURL` i `AllowedIP`.

Domyślny plik konfiguracyjny (`zabbix_mcp_server.conf`) znajduje się w katalogu `/etc/zabbix/`; odmawia on dostępu do narzędzi, które najczęściej ujawniają dane wrażliwe. Aby włączyć narzędzie, usuń odpowiednią linię `DenyTool`.

Zobacz także: Najlepsze praktyki dotyczące bezpieczeństwa

Uruchamianie Aby uruchomić serwer MCP, wykonaj:

```
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf
```

Aby sprawdzić plik konfiguracyjny i zakończyć działanie bez uruchamiania serwera, użyj opcji `-T`:

```
zabbix_mcp_server -c /etc/zabbix/zabbix_mcp_server.conf -T
```

Obsługiwane są następujące opcje wiersza poleceń:

Opcja	Opis
`-c <file>`	Ścieżka do pliku konfiguracyjnego. Domyślnie jest to `/etc/zabbix/zabbix_mcp_server.conf`.
`-T`	Sprawdź poprawność pliku konfiguracyjnego i zakończ działanie.
`-h`	Wyświetl informacje pomocy.
`-V`	Wyświetl numer wersji.

Jeśli serwer MCP został zainstalowany ze źródeł, jednostkę systemd należy utworzyć ręcznie, na przykład:

```
[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
```

Łączenie klienta MCP Każdy klient MCP obsługujący transport Streamable HTTP może zostać połączony z serwerem MCP. Wymagane są następujące ustawienia:

Ustawienie	Wartość
Transport	Streamable HTTP
URL	`https://<host>:<ListenPort>/mcp`
Nagłówek	`Authorization: Bearer <Zabbix API token>`, wysyłany z każdym żądaniem

Każdy użytkownik musi skonfigurować własny token API.

Aby zweryfikować konfigurację, wyślij pojedyncze żądanie do serwera 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"}'
```

Kod stanu HTTP 200 oraz lista dostępnych narzędzi w odpowiedzi potwierdzają, że serwer MCP jest osiągalny i akceptuje token; lista pokazuje, które narzędzia są udostępnione. Inne kody stanu wskazują następujące problemy:

- 400 - nagłówek `Accept` nie zawiera obu typów multimediów wymaganych przez transport;
- 401 - brakuje nagłówka `Authorization` z tokenem;
- 403 - adres klienta nie znajduje się na liście w parametrze `AllowedIP`.

Zatwierdzanie i dostęp do narzędzi Jeśli klient MCP obsługuje wywoływanie formularzy, każda zmiana musi zostać zatwierdzona przez użytkownika przed wysłaniem jej do Zabbix. Klient MCP pokazuje użytkownikowi, co zostanie zmienione, a zmiana jest stosowana tylko wtedy, gdy użytkownik ją potwierdzi; jeśli użytkownik odrzuci lub zamknie żądanie, w Zabbix nie zostaną wprowadzone żadne zmiany. Zatwierdzenie jest ważne przez pięć minut i dotyczy wyłącznie dokładnie pokazanej zmiany; w tym czasie można je wykorzystać ponownie.

Jeśli klient MCP nie obsługuje wywoływania formularzy (w tym klientów obsługujących tylko wywoływanie adresów URL), zmiany są stosowane natychmiast, bez zatwierdzenia. W takim przypadku należy ograniczyć narzędzia zapisu i uprawnienia API Zabbix.

Zatwierdzanie jest szczególnie ważne, ponieważ dane wyjściowe narzędzi mogą zawierać niezaufany tekst z monitorowanych systemów, który może wpływać na agenta.

Użyj AllowTool, DenyTool, AllowToolRegexp i DenyToolRegexp, aby kontrolować, które narzędzia serwer MCP udostępnia. Zablokowanych narzędzi nie można wywoływać ani wyświetlać, a powiązane z nimi zasoby nie są dostępne.

Warning:
Reguły AllowTool, DenyTool, AllowToolRegexp i DenyToolRegexp ograniczają dostęp MCP, ale nie zmieniają uprawnień tokenu Zabbix. Jeśli adres frontend zostanie ujawniony agentowi AI, agent może próbować wysyłać bezpośrednie wywołania API. Należy uniemożliwić agentowi dostęp do lokalizacji frontend Zabbix i konfiguracji serwera MCP, aby żądania musiały przechodzić przez serwer MCP i jego mechanizmy kontroli bezpieczeństwa.

Najlepsze praktyki w zakresie bezpieczeństwa Tokeny API

Uwierzytelnianie i autoryzacja są w całości obsługiwane przez Zabbix. Klient MCP wysyła token API Zabbix z każdym żądaniem; serwer MCP przekazuje go bez przechowywania ani rejestrowania, natomiast Zabbix weryfikuje token, egzekwuje uprawnienia i rejestruje działania w dzienniku audytu.

Dla każdego agenta i użytkownika używaj osobnego tokenu z minimalnymi uprawnieniami i określonym terminem wygaśnięcia. Przechowuj tokeny w bezpiecznej lokalizacji i regularnie je zmieniaj.

Szyfrowanie TLS

Aby zapobiec przesyłaniu tokenów w postaci jawnego tekstu, ustaw `TLSAccept=cert` i określ certyfikat oraz klucz (`TLSCertFile` i `TLSKeyFile`) albo uruchom serwer MCP za zaufanym odwrotnym proxy kończącym połączenie TLS i ustaw `TLSAccept=unencrypted`. Nieszyfrowany protokół HTTP jest akceptowalny wyłącznie w zaufanej sieci.

Ograniczenia dostępu

Ogranicz parametr `AllowedIP` do adresów zaufanych klientów MCP i używaj uprawnień Zabbix, aby ograniczyć zakres działań tokenów API.

serwer MCP nie zapewnia ograniczania częstotliwości żądań. Każde żądanie MCP zwykle skutkuje jednym wywołaniem API Zabbix, jednak agent działający w pętli nadal może generować znaczne obciążenie.

Ochrona danych wrażliwych

Klucze pozycji mogą zawierać dane uwierzytelniające (na przykład w pozycjach SSH, JMX, ODBC i HTTP agent), wartości makr mogą zawierać hasła i ciągi community, interfejsy ujawniają adresy wewnętrzne, a wartości pozycji typu log lub tekst mogą zawierać dowolne dane. Z tego powodu narzędzia `item_get`, `lld_get`, `interface_get`, `macro_get` i `history_get` są domyślnie zablokowane w dostarczonym pliku konfiguracyjnym.

Aby uniknąć ujawnienia wrażliwych wartości makr dostawcy LLM podłączonego agenta, użyj makr tajnych; ich wartości nigdy nie są zwracane przez API.

Narzędzia i zasoby

Przegląd

Serwer MCP Zabbix udostępnia funkcje Zabbix klientom MCP jako *narzędzia* i *zasoby*.

Nazwy narzędzi opisują ich działanie. Jeśli ustawiono parametr ToolPrefix, prefiks jest dodawany do każdej nazwy narzędzia; reguły dostępu do narzędzi odnoszą się do zwykłych nazw wymienionych poniżej.

Narzędzia odczytu

Narzędzie	Metoda Zabbix API	Argumenty	Sortowanie według	Zasób
`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`	interface ID	-
`macro_get`	usermacro.get	`hostids` lub `global`, `macro_names`, `limit`	macro	-
`problem_get`	problem.get	`eventids`, `groupids`, `hostids`, `severities`, `acknowledged`, `suppressed`, `recent`, `limit`	event ID, malejąco	-
`maintenance_get`	maintenance.get	`maintenanceids`, `groupids`, `hostids`, `limit`	name	-
`history_get`	history.get	`itemids`, `time_from`, `time_till`, `type`, `limit`	clock, malejąco	-
`trigger_get`	trigger.get	`triggerids`, `hostids`, `search`, `limit`	description	`zabbix://trigger/<id>`

Należy pamiętać, że:

- `host_get` i `template_get` zwracają również wyzwalacze każdego obiektu (ograniczony zestaw pól), tagi i połączone szablony. Pozycje, reguły wykrywania, interfejsy i makra są zwracane przez odpowiednie narzędzia.
- `history_get` wymaga zakresu czasu nie szerszego niż 7 dni, rozpoczynającego się nie wcześniej niż 7 dni temu. Argument `type` musi mieć jedną z wartości `float`, `unsigned`, `character`, `text`, `log` lub `json` i musi odpowiadać typowi wartości pozycji; jeśli typ jest nieprawidłowy, zwracany jest pusty wynik, którego nie można odróżnić od pozycji bez danych. Historii pozycji binarnych (typ wartości 5) nie można pobrać za pośrednictwem serwera MCP.
- Narzędzia odczytu żądają stałej listy pól; nigdy nie są żądane wszystkie pola (`output: extend`).
- Wyniki narzędzi zawierających zasób mają pole `uri`, dzięki czemu agent może przejść od listy do pojedynczej encji.

Format odpowiedzi

Wszystkie narzędzia odczytu zwracają odpowiedź w tym samym formacie:

```
{
  "results": [ ... ],
  "count": 2,
  "truncated": true,
  "truncated_by": "count",
  "limit": 2,
  "note": "Osiągnięto limit liczby wyników ..."
}
```

Odpowiedź jest ograniczana przez argument `limit` (domyślnie: 100, maksymalnie: 1000) oraz przez serializowany rozmiar odpowiedzi (MaxResponseSize, domyślnie: 64K). Odpowiedź zostaje zatrzymana po osiągnięciu któregokolwiek z limitów. Odpowiedź jest skracana wyłącznie między wynikami, nigdy wewnątrz wyniku, a skrócona odpowiedź wskazuje, który limit został osiągnięty. Każde narzędzie ma stałą kolejność sortowania, dlatego skrócona odpowiedź zawsze zawiera początek pełnego zestawu wyników. Stronicowanie nie jest obsługiwane; aby uzyskać inne wyniki, zawęź zapytanie.

Narzędzia problemów

Wszystkie narzędzia problemów wywołują metodę event.acknowledge z odpowiednimi flagami akcji. Argument `eventids` jest wymagany; identyfikatory zdarzeń można pobrać za pomocą `problem_get`. Za pomocą dowolnego narzędzia problemów można dodać wiadomość (komentarz). Argument `message` jest opcjonalny, z wyjątkiem `problem_annotate`, dla którego jest wymagany; jeśli zostanie określony, wiadomość zostanie zarejestrowana w tym samym wywołaniu Zabbix API.

Tool	Effect	Additional arguments
`problem_acknowledge`	Potwierdzenie problemu.	-
`problem_unacknowledge`	Usunięcie potwierdzenia.	-
`problem_close`	Ręczne zamknięcie problemu (tylko w przypadku wyzwalaczy, które umożliwiają ręczne zamykanie).	-
`problem_annotate`	Zarejestrowanie wiadomości.	`message` (wymagany)
`problem_change_severity`	Zmiana ważności problemu.	`severity` (0-5)
`problem_suppress`	Wyciszenie problemu do określonego czasu lub bezterminowo.	`suppress_until`
`problem_unsuppress`	Usunięcie wyciszenia.	-

Nie można tworzyć ani usuwać problemów; wszystkie narzędzia problemów aktualizują istniejące problemy.

Narzędzia do obsługi okresów utrzymania

Narzędzie	Metoda API Zabbix	Argumenty
`maintenance_create`	maintenance.create	`name`, `groupids` i/lub `hostids`, `active_since`, `active_till`, `timeperiods`, `description`, `tags`, `tags_evaltype`
`maintenance_update`	maintenance.update	`maintenanceid` oraz dowolne z powyższych argumentów. Argumenty będące listami zastępują cały zestaw wartości; pusta lista czyści zestaw.
`maintenance_delete`	maintenance.delete	`maintenanceids`

Po zaktualizowaniu okresu utrzymania argumenty będące listami (takie jak `hostids`, `groupids` i `timeperiods`) zastępują istniejące wartości, zamiast je dodawać. Na przykład jeśli okres utrzymania obejmuje hosty A i B, a wywołanie `maintenance_update` zawiera `hostids` tylko z hostem C, okres utrzymania będzie obejmować wyłącznie hosta C.

`maintenance_create` jest jedyną operacją, która nie jest idempotentna, a `maintenance_delete` jest jedyną operacją, której nie można cofnąć. Aby uniemożliwić ich używanie, zablokuj te narzędzia za pomocą parametru DenyTool.

Działanie narzędzi zapisu

Narzędzia problemów i konserwacji są *narzędziami zapisu*. Działają w następujący sposób:

- Argumenty są sprawdzane przed wykonaniem jakiegokolwiek wywołania API Zabbix; jeśli sprawdzanie zakończy się niepowodzeniem, błąd wskazuje niespełnione ograniczenie.
- Jeśli klient obsługuje tę funkcję, zmiana jest stosowana dopiero po jej zatwierdzeniu przez użytkownika.
- Zmiana jest wykonywana przez Zabbix przy użyciu tokenu wywołującego. Zabbix sprawdza uprawnienia i rejestruje zmianę w dzienniku audytu w odniesieniu do właściciela tokenu.
- Błędy Zabbix, w tym błędy uprawnień, są zwracane bez zmian jako obiekt błędu z polami `code`, `message` i `data`.
- Narzędzia zapisu zwracają bezpośrednio wynik Zabbix; ograniczenia odpowiedzi narzędzi odczytu nie mają zastosowania.
- Wywołania zapisu nigdy nie są ponawiane przez serwer MCP.

Zasoby

Zasoby używają schematu URI `zabbix://<kind>/<id>`. Tylko `zabbix://about` jest uwzględniany w odpowiedzi `resources/list`; identyfikatorów encji nie można wyliczyć i są one uzyskiwane z wyników działania narzędzi. Odpowiedź `resources/templates/list` zawiera listę rodzajów encji.

URI	Zawartość
`zabbix://about`	URL API, URL frontendu, wersja Zabbix, tożsamość, na którą rozwiązuje się token wywołującego, oraz lista udostępnionych narzędzi.
`zabbix://host/<hostid>`	Rekord hosta bez pozycji, wyzwalaczy, interfejsów i makr.
`zabbix://hostgroup/<groupid>`	Rekord grupy hostów bez listy członków.
`zabbix://template/<templateid>`	Rekord szablonu bez pozycji, wyzwalaczy i reguł wykrywania.
`zabbix://templategroup/<groupid>`	Rekord grupy szablonów bez listy członków.
`zabbix://item/<itemid>`	Definicja pozycji bez bieżącej wartości.
`zabbix://trigger/<triggerid>`	Definicja wyzwalacza bez hostów, zależności, funkcji i znaczników.

Zasób, któremu odpowiada odmówione narzędzie, nie istnieje. Nieistniejąca encja oraz encja, do której właściciel tokenu nie ma dostępu, są w obu przypadkach zgłaszane jako *resource not found*; przypadki te są celowo nierozróżnialne. Na zasób `zabbix://about` nie mają wpływu reguły dostępu do narzędzi.

Parametry konfiguracji: serwer Zabbix MCP

Przegląd

serwer MCP Zabbix to proces umożliwiający agentom AI pracę z Zabbix za pomocą protokołu Model Context Protocol (MCP).

W tej sekcji wymieniono parametry obsługiwane przez plik konfiguracyjny serwera MCP Zabbix (`zabbix_mcp_server.conf`).

Parametry są wymienione bez dodatkowych informacji. Kliknij parametr, aby wyświetlić pełne szczegóły.

Parametr	Opis
AllowedIP	Lista adresów IP klientów MCP rozdzielonych przecinkami, opcjonalnie w notacji CIDR, lub nazw DNS klientów MCP.
AllowTool	Reguła wieloznaczna zezwalająca na użycie narzędzia.
AllowToolRegexp	Reguła wyrażenia regularnego zezwalająca na użycie narzędzi.
DebugLevel	Poziom debugowania.
DenyTool	Reguła wieloznaczna odmawiająca użycia narzędzia.
DenyToolRegexp	Reguła wyrażenia regularnego odmawiająca użycia narzędzi.
FrontendURL	Punkt końcowy API Zabbix.
IgnoreURLCertErrors	Określa sposób obsługi błędów walidacji certyfikatu TLS podczas uzyskiwania dostępu do API Zabbix.
ListenIP	Adres IP, na którym nasłuchiwać.
ListenPort	Port, na którym nasłuchiwać.
LogFile	Nazwa pliku dziennika.
LogFileSize	Maksymalny rozmiar pliku dziennika.
LogType	Typ danych wyjściowych dziennika.
MaxResponseSize	Maksymalny rozmiar odpowiedzi zwracanej przez narzędzie odczytu.
PidFile	Nazwa pliku PID.

Parametr	Opis
Timeout	Maksymalny czas (w sekundach) oczekiwania na odpowiedź z API Zabbix.
TLSAccept	Typ akceptowanych połączeń przychodzących.
TLSCertFile	Pełna ścieżka do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów.
TLSKeyFile	Pełna ścieżka do pliku zawierającego klucz prywatny serwera.
ToolPrefix	Prefiks dodawany do nazw wszystkich narzędzi.

Wszystkie parametry są opcjonalne, chyba że wyraźnie zaznaczono, że dany parametr jest wymagany.

Należy pamiętać, że:

- Wartości domyślne odzwierciedlają wartości domyślne procesu, a nie wartości w dostarczonych plikach konfiguracyjnych;
- Wartości obsługują zmienne środowiskowe w formacie `${NAME}`;
- Zabbix obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 bez BOM;
- Komentarze rozpoczynające się od # są obsługiwane tylko na początku wiersza.

Szczegóły parametrów

AllowedIP

Lista rozdzielanych przecinkami adresów IP, opcjonalnie w notacji CIDR, lub nazw DNS klientów MCP. Połączenia przychodzące będą akceptowane wyłącznie z hostów wymienionych na tej liście.

Przykład:

```
AllowedIP=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,mcp-client.example.com
```

Wymagane: tak

AllowTool

Reguła z symbolem wieloznacznym zezwalająca na użycie narzędzia. Parametr można określić wielokrotnie.
 Wzorce z symbolami wieloznacznymi używają składni klucza pozycji, w której * oznacza dowolne znaki. Reguła jest dopasowywana do zwykłej nazwy narzędzia i ignoruje ToolPrefix.
 Wszystkie reguły `AllowTool`, `DenyTool`, `AllowToolRegexp` i `DenyToolRegexp` tworzą jedną listę w kolejności, w jakiej zostały określone; pierwsza pasująca reguła decyduje o wyniku. Narzędzie, do którego nie pasuje żadna reguła, jest dozwolone.

Przykład:

```
AllowTool=host_get
```

AllowToolRegexp

Reguła wyrażenia regularnego zezwalająca na używanie narzędzi. Parametr można określić wielokrotnie.
 Wyrażenia regularne używają składni RE2 i dopasowują dowolną część zwykłej nazwy narzędzia, chyba że zastosowano kotwice.

Przykład:

```
AllowToolRegexp=^(host|hostgroup|trigger|problem)_get$
```

DebugLevel

Określ poziom debugowania: - *0* - podstawowe informacje o uruchamianiu i zatrzymywaniu procesów Zabbix; - *1* - informacje krytyczne; - *2* - informacje o błędach; - *3* - ostrzeżenia; - *4* - debugowanie (generuje dużą ilość informacji); - *5* - rozszerzone debugowanie (generuje jeszcze więcej informacji).

Wartość domyślna: 3
 Zakres: 0-5

DenyTool

Reguła z symbolem wieloznacznym, która blokuje narzędzie. Parametr można określić wielokrotnie.
 Zablokowane narzędzie nie jest uwzględniane na liście narzędzi i nie można go wywołać. Szczegółowe informacje zawiera sekcja AllowTool.

Przykład:

```
DenyTool=maintenance_delete
```

DenyToolRegexp

Reguła wyrażenia regularnego, która blokuje narzędzia. Parametr można określić wielokrotnie.
 Szczegółowe informacje zawiera sekcja AllowToolRegexp.

Przykład:

```
DenyToolRegexp=^problem_(close|suppress)$
```

FrontendURL

Punkt końcowy API Zabbix: adres URL frontend Zabbix kończący się ciągiem `api_jsonrpc.php`.

Przykład:

`FrontendURL=https://zabbix.example.com/api_jsonrpc.php`

Wymagane: tak

IgnoreURLCertErrors

Określa sposób obsługi błędów walidacji certyfikatu TLS podczas uzyskiwania dostępu do API Zabbix:
 false - nie ignoruj błędów certyfikatu;
 true - ignoruj błędy certyfikatu (używaj tylko w środowisku deweloperskim).

Wartość domyślna: `false`

ListenIP

Adres IP, na którym nasłuchiwać.

Wartość domyślna: `0.0.0.0`

ListenPort

Port, na którym nasłuchuje. Ścieżka punktu końcowego MCP to `/mcp`.

Wartość domyślna: 8443
 Zakres: 1024-32767

LogFile

Nazwa pliku dziennika.

Wartość domyślna: `/tmp/zabbix_mcp_server.log`

Wymagane: Nie

LogFileSize

Maksymalny rozmiar pliku dziennika w MB.
0 - wyłącza automatyczną rotację dziennika.

Wartość domyślna: 1
 Zakres: 0-1024

LogType

Typ danych wyjściowych dziennika: - *file* - zapisuje dziennik w pliku określonym przez parametr LogFile; - *system* - zapisuje dziennik w syslog; - *console* - zapisuje dziennik na standardowym wyjściu.

Wartość domyślna: `file`

MaxResponseSize

Maksymalny rozmiar odpowiedzi zwracanej przez narzędzie odczytu. Osiągnięcie tego limitu lub wartości argumentu `limit` narzędzia — w zależności od tego, co nastąpi wcześniej — zatrzymuje odpowiedź.
 Obsługuje przyrostki jednostek.

Wartość domyślna: 64K
 Zakres: 16K-256K

PidFile

Nazwa pliku PID.

Wartość domyślna: `/tmp/zabbix_mcp_server.pid`

Limit czasu

Maksymalny czas (w sekundach) oczekiwania na odpowiedź z API Zabbix.

Wartość domyślna: 10
 Zakres: 3-30

TLSAccept

Jakie połączenia przychodzące akceptować: - *unencrypted* - akceptowanie połączeń bez szyfrowania (domyślnie); - *cert* - akceptowanie połączeń TLS z certyfikatem.

Wartość domyślna: `unencrypted`

TLSCertFile

Pełna ścieżka do pliku zawierającego certyfikat serwera lub łańcuch certyfikatów.

Wymagane: Tak, jeśli parametr `TLSAccept` jest ustawiony na `cert`; w przeciwnym razie nie

TLSKeyFile

Pełna ścieżka do pliku zawierającego klucz prywatny serwera.

Wymagane: Tak, jeśli parametr `TLSAccept` ma wartość `cert`; w przeciwnym razie nie

ToolPrefix

Prefiks dodawany do nazw wszystkich narzędzi, na przykład w celu uniknięcia konfliktów nazw z narzędziami innych serwerów MCP. Na przykład przy ustawieniu `ToolPrefix=zbx_` narzędzie `host_get` jest udostępniane jako `zbx_host_get`.
 Dozwolone znaki: litery, cyfry, podkreślenie (_), łącznik (-) i kropka (.).
 Reguły dostępu do narzędzi odnoszą się do nazw narzędzi bez prefiksu.

Przykład:

`ToolPrefix=zbx_`

Zmienne środowiskowe

Omówienie Zmienne środowiskowe umożliwiają konfigurację komponentów Zabbix bez wpisywania na sztywno wartości w plikach konfiguracyjnych. Ułatwia to zarządzanie konfiguracjami w dynamicznych środowiskach, takich jak Docker, gdzie zmienne mogą być przekazywane w czasie uruchomienia, aby dostosować je do różnych konfiguracji.

W najprostszym przypadku, ustawiając wartość parametru konfiguracyjnego DebugLevel serwera Zabbix na zmienną środowiskową, można następnie użyć jej do skonfigurowania serwera podczas uruchamiania:

```
### Plik konfiguracyjny serwera Zabbix:
DebugLevel=${NEW_DEBUG_LEVEL}

### Uruchamianie serwera Zabbix:
NEW_DEBUG_LEVEL=5 /usr/sbin/zabbix_server
```

Zmienne środowiskowe są obsługiwane przez następujące komponenty Zabbix:

- Serwer
- Proxy
- Agent (UNIX lub Windows)
- Agent 2 (UNIX lub Windows), w tym wtyczki
- Usługa web
- Zabbix sender (przy użyciu opcji -c, --config)

Ważne uwagi

- Gdy parametr konfiguracji jest ustawiony na zmienną środowiskową, która nie została określona podczas uruchamiania komponentu, używana jest domyślna wartość parametru.
- Podczas korzystania z poleceń czasu działania (np. w celu zwiększenia poziomu logowania agent), należy podać wszystkie wcześniej używane zmienne środowiskowe. Wynika to z faktu, że komponenty Zabbix używają pliku konfiguracyjnego do wykonywania poleceń czasu działania; jeśli zmienne środowiskowe zostaną pominięte, zostaną użyte domyślne wartości parametrów konfiguracji. Zobacz Przykłady.
- Polecenie czasu działania `userparameter_reload` polecenie czasu działania nie obsługuje ponownego wczytywania zmiennych środowiskowych. Podczas ponownego wczytywania zmienne są ignorowane, a wczytywane są tylko parametry o zwykłych wartościach.
- Bieżące zmienne środowiskowe procesu, które były używane w plikach konfiguracyjnych, są czyszczone po uruchomieniu komponentu Zabbix. Zapewnia to, że procesy potomne (np. zdalne skrypty wykonywane przez Zabbix) nie mogą uzyskać dostępu do tych zmiennych. Należy jednak pamiętać, że początkowe zmienne procesu nadal można odczytać (np. za pomocą pliku `/proc/<PID>/environ`).

Składnia Zmienne środowiskowe muszą używać następującej składni: `${alfanumeryczne/podkreślenia}`.

Nazwa zmiennej może zawierać tylko litery (a-z, A-Z), podkreślenia (_), oraz cyfry (0-9) i nie może zaczynać się od cyfry.

Zmienne, które nie odpowiadają wymaganej składni lub są połączone ze zwykłą wartością, będą traktowane jako zwykłe wartości, co może powodować błędy.

Poprawna składnia zmiennej:

```
DebugLevel=${NEW_DEBUG_LEVEL}
Hostname=${ZBX_HOSTNAME}
LogFile=${LogFile_001}
```

Niepoprawna składnia zmiennej:

```
DebugLevel=${5_DebugLevel}
Hostname=${ZBX.HOSTNAME 1}
LogFile=/${HOME}/zabbix/zabbix_server.log
```

> **Note:**
> W systemie Windows nazwy zmiennych środowiskowych nie rozróżniają wielkości liter.

Przykłady Poniższe przykłady pokazują, jak skonfigurować i używać zmiennych środowiskowych z komponentami Zabbix.

Przykład 1: Konfigurowanie i testowanie agenta Zabbix

1. Ustaw zmienne środowiskowe w pliku konfiguracyjnym agenta:

```
Hostname=${ZBX_HOSTNAME}
ServerActive=${ServerActive}
```

2. Przetestuj plik konfiguracyjny:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

3. Uruchom agenta ze zmiennymi środowiskowymi:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Podczas korzystania z poleceń wykonywanych w czasie działania (np. w celu zwiększenia poziomu logowania agenta) należy określić wszystkie wcześniej używane zmienne środowiskowe:

```
ZBX_HOSTNAME="New Zabbix agent" ServerActive=127.0.0.1 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agent
```

Dzieje się tak, ponieważ agent używa swojego pliku konfiguracyjnego do wykonywania poleceń wykonywanych w czasie działania; jeśli zmienne środowiskowe zostaną pominięte, zostaną użyte wartości domyślne parametrów konfiguracyjnych.

Alternatywnie, po ustawieniu zmiennych środowiskowych w pliku konfiguracyjnym agenta można udostępnić je procesom (np. za pomocą polecenia `export`). Zmniejsza to ryzyko nieoczekiwanego działania spowodowanego brakiem zmiennych lub ich nieprawidłowym ustawieniem.

```
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
```

Przykład 2: Konfigurowanie agent dla kontenera

Jeśli tworzysz i konfigurujesz własny niestandardowy obraz dla komponentów Zabbix (np. agent), możesz zdefiniować parametry konfiguracji za pomocą zmiennych środowiskowych, a następnie uruchomić kontener z tymi zmiennymi.

1. Podczas przygotowywania obrazu ustaw zmienne środowiskowe w pliku konfiguracyjnym agent:

```
Hostname=${ZBX_HOSTNAME}
BufferSize=${BUFSZ}
ListenPort=${LISTENPORT}
UserParameter=${_UsrPar01}
UserParameter=${_UsrPar02}
```

2. Po zbudowaniu obrazu kontenera uruchom kontener agent (np. Docker) ze zmiennymi środowiskowymi:

```
docker run --name my-zabbix-agent -e ZBX_HOSTNAME="new-hostname" -e BUFSZ=1000 -e LISTENPORT=20050 -e _Usr
```

3. W przypadku korzystania z poleceń wykonywanych w czasie działania (np. w celu zwiększenia poziomu rejestrowania agent) uzyskaj dostęp do powłoki kontenera i wykonaj polecenie wykonywane w czasie działania:

```
docker exec -it <containerid> sh
/usr/sbin/zabbix_agentd -R log_level_increase
```

Attention:
Polecenie wykonywane w czasie działania `userparameter_reload` nie obsługuje ponownego ładowania zmiennych środowiskowych. Podczas ponownego ładowania zmienne są ignorowane, a ponownie ładowane są tylko parametry ze zwykłymi wartościami.

Konfiguracja

Omówienie Konfiguracja w Zabbix polega na określeniu, które hosty i systemy mają być monitorowane, zdefiniowaniu, jakie dane mają być zbierane, oraz ustaleniu, w jaki sposób mają być dostarczane powiadomienia w przypadku wystąpienia problemów.

Użyj paska bocznego, aby przejść do odpowiednich obszarów konfiguracji.

Kluczowe zadania konfiguracyjne

- Skonfiguruj nowy host lub grupę hostów → Hosts and host groups
- Określ, jakie dane mają być zbierane z hostów → Items
- Utwórz reguły wykrywania problemów → Problem detection with triggers
- Skonfiguruj sposób dostarczania powiadomień → Notifications upon events
- Wyświetlaj dane monitorowania za pomocą pulpitów, wykresów i map → Visualization
- Ponownie wykorzystuj konfiguracje w wielu hostach → Templates and template groups, Templates out of the box
- Kontroluj dostęp użytkowników i uprawnienia → Users and user groups

Najlepsze praktyki konfiguracji Wybory konfiguracyjne wpływają na wydajność i łatwość utrzymania. Chociaż każda konfiguracja Zabbixa jest inna, poniższe najlepsze praktyki można wykorzystać jako podstawę i dostosować je w razie potrzeby.

Oddzielny host dla każdej monitorowanej instancji

Host w Zabbiksie jest bytem logicznym, a nie fizyczną maszyną ani urządzeniem. Utwórz oddzielny host dla każdej niezależnej instancji, takiej jak maszyna wirtualna, baza danych, kontener lub przełącznik sieciowy.

Takie podejście zapewnia:

1. Czystsze dane monitorowania dzięki oddzielnym pozycjom, wyzwalaczom i powiadomieniom alarmowym dla każdego hosta.
2. Precyzyjną kontrolę dostępu użytkowników za pomocą ról użytkowników zgodną z zasadą najmniejszych uprawnień.

Użyj szablonów i masowej aktualizacji dla podobnych hostów

Dla podobnych hostów, takich jak *Network switch 1* i *Network switch 2*, możesz utworzyć szablon z wymaganymi metrykami (*Network switch template*) i pogrupować hosty w grupę hostów (*Network switches*).

Podczas dodawania nowych przełączników możesz użyć masowej aktualizacji w *Data Collection* > *Hosts*, aby powiązać szablon zamiast ręcznie klonować hosty.

Skonfiguruj pozycje nadrzędne i zależne, aby zminimalizować liczbę żądań

Aby zminimalizować liczbę żądań do obiektu docelowego, skonfiguruj pozycję nadrzędną tak, aby zbierała dane w jednym żądaniu, a następnie użyj pozycji zależnych z przetwarzaniem wstępnym do wyodrębnienia określonych wartości.

Na przykład pozycja nadrzędna może zbierać odpowiedź JSON lub XML z wieloma metrykami albo wykonywać zapytanie do bazy danych zwracające wiele kolumn (np. liczbę otwartych połączeń, liczbę przerwanych połączeń, maksymalną dozwoloną liczbę jednoczesnych połączeń oraz łączną skumulowaną liczbę połączeń od uruchomienia), a pozycje zależne będą analizować i zapisywać każdą wartość osobno.

Odrzuć historię pozycji nadrzędnej po zebraniu danych i zachowuj tylko dane pozycji zależnych.

Wdróż proxy Zabbix, gdy jest to potrzebne

Jeśli wszystkie hosty znajdują się w tej samej sieci lokalnej co serwer Zabbix i nie występują problemy ze skalowalnością ani wydajnością, proxy może nie być potrzebny. W większych lub bardziej złożonych środowiskach proxy umożliwia bardziej równomierne rozłożenie obciążenia.

Dodaj proxy Zabbix, gdy:

1. Monitorujesz wiele hostów przy użyciu różnych metod zbierania metryk za zaporą sieciową. Proxy zbiera dane z hostów i przekazuje je do serwera Zabbix, zmniejszając liczbę otwartych portów zapory sieciowej.

2. Monitorujesz zdalne lokalizacje, oddziały lub sieci. Proxy Zabbix w zdalnych lokalizacjach kontynuują zbieranie danych podczas przerw w działaniu sieci i wysyłają dane po przywróceniu połączenia.
3. Zarządzasz wdrożeniami na dużą skalę. Jeśli masz wiele hostów i/lub zbierasz dużą liczbę wartości na sekundę, wdróż proxy, aby zmniejszyć obciążenie serwera i poprawić wydajność.

Konfigurowanie niezależnego pobierania tajnych makr

Możesz używać makr użytkownika secret jako tajnego tekstu lub tajnych makr skarbca.

Domyślnie wartości tajnych makr są pobierane przez serwer Zabbix i propagowane do proxy. Aby zwiększyć bezpieczeństwo w przypadku tajnych makr skarbca, skonfiguruj wartości makr, tak aby były pobierane niezależnie przez serwer Zabbix i proxy.

Konfigurowanie szablonu

Przegląd

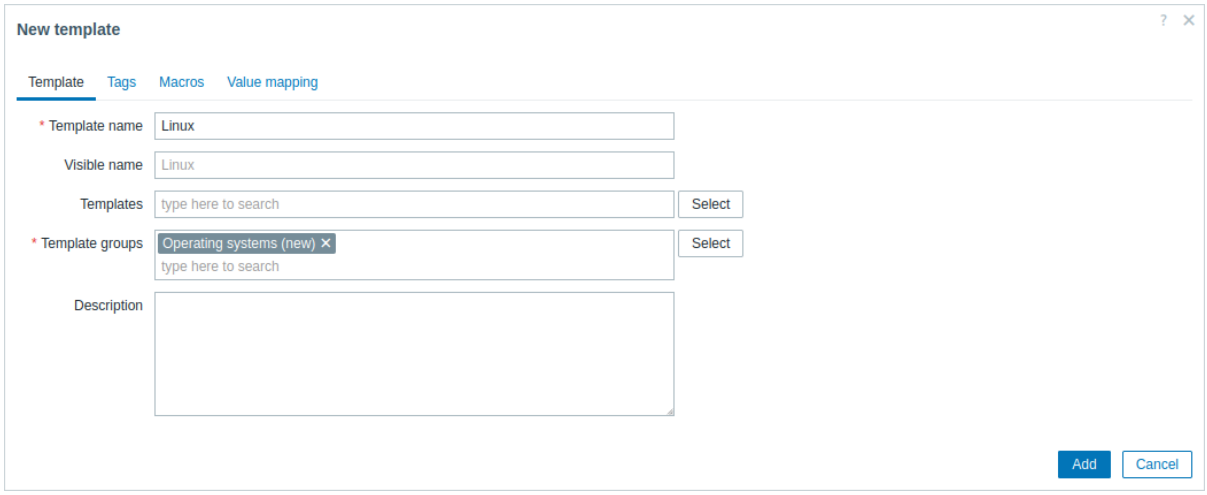
Konfiguracja szablonu wymaga najpierw utworzenia szablonu poprzez zdefiniowanie jego ogólnych parametrów, a następnie dodania do niego encji (pozycji, wyzwalaczy, wykresów itp.).

Tworzenie szablonu

Aby utworzyć szablon:

1. Przejdź do *Gromadzenie danych* > *Szablony*.
2. Kliknij *Utwórz szablon*.
3. Edytuj atrybuty szablonu.

Karta **Szablon** zawiera ogólne atrybuty szablonu.



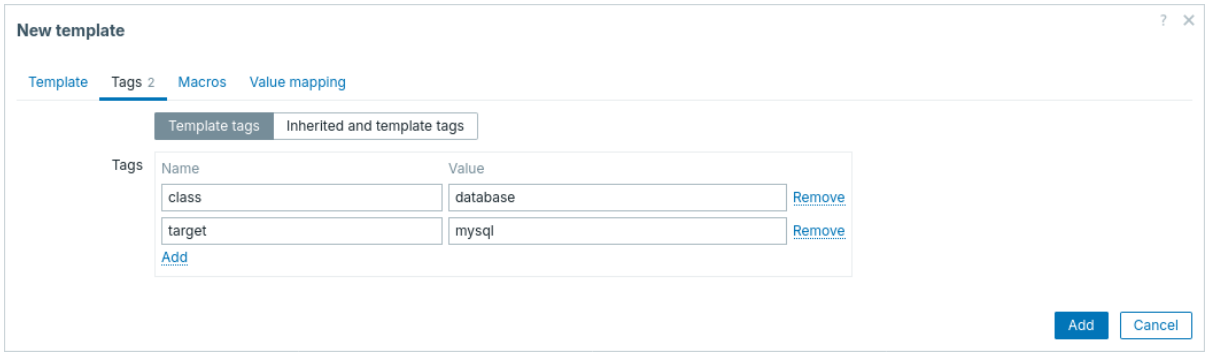
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Atrybuty szablonu:

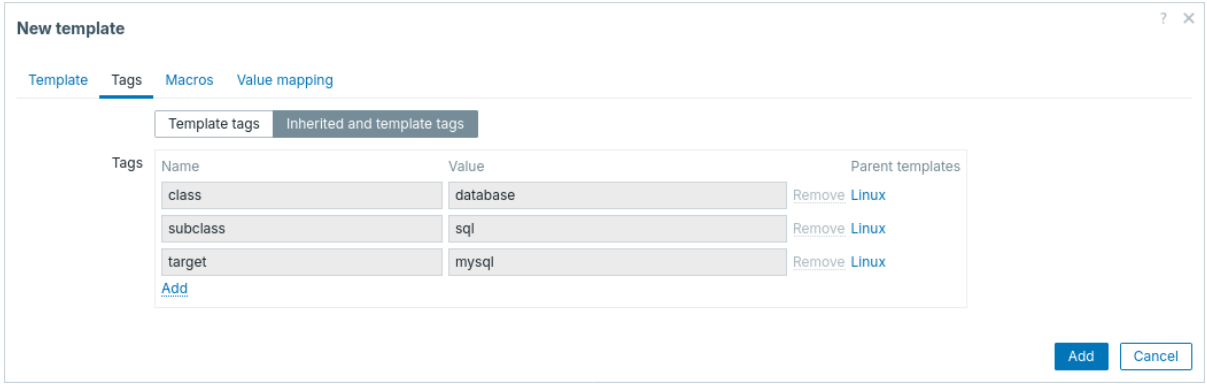
Parametr	Opis
Nazwa szablonu	Unikalna nazwa szablonu. Dozwolone są znaki alfanumeryczne, spacje, kropki, myślniki i podkreślenia. Spacje na początku i na końcu nie są dozwolone.
Nazwa widoczna	Jeśli ustawisz tę nazwę, będzie ona wyświetlana na listach, mapach itd.

Parametr	Opis
Szablony	Połącz jeden lub więcej szablonów z tym szablonem. Wszystkie encje (pozycje, wyzwalacze itd.) zostaną odziedziczone z połączonych szablonów. Aby połączyć nowy szablon, wpisz jego nazwę w polu *Szablony* — zostanie wyświetlona lista rozwijana pasujących szablonów. Możesz również kliknąć *Wybierz*, aby otworzyć okno podręczne *Szablony*, które początkowo nie zawiera żadnych szablonów. Aby wypełnić listę, wpisz nazwę grupy szablonów w polu *Grupa szablonów* (i wybierz ją z sugerowanej listy) lub kliknij przycisk *Wybierz* obok pola *Grupa szablonów*, aby otworzyć okno podręczne *Grupy szablonów*. Po wybraniu grupy szablonów okno podręczne *Szablony* zostanie zaktualizowane i wyświetli szablony należące do tej grupy. Wybrane szablony zostaną połączone podczas zapisywania lub aktualizowania konfiguracji. Aby odłączyć szablon, użyj jednej z dwóch opcji w bloku *Szablony*: *Odłącz* — odłącza szablon, ale zachowuje jego encje (pozycje, wyzwalacze itd.); *Odłącz i wyczyść* — odłącza szablon i usuwa wszystkie jego encje (pozycje, wyzwalacze itd.).
Grupy szablonów	Grupy szablonów, do których należy szablon.
Opis	Opis szablonu.
Dostawca i wersja	Dostawca i wersja szablonu; wyświetlane tylko podczas aktualizowania istniejących szablonów (szablonów gotowych do użycia dostarczonych przez Zabbix, zaimportowanych szablonów lub szablonów zmodyfikowanych za pośrednictwem Template API), jeśli konfiguracja szablonu zawiera takie informacje. Nie można ich modyfikować w frontend Zabbix. W przypadku szablonów gotowych do użycia wersja jest wyświetlana w następujący sposób: główna wersja Zabbix, separator (”-”), numer rewizji (zwiększany przy każdej nowej wersji szablonu i resetowany przy każdej głównej wersji Zabbix). Na przykład 7.0-0, 7.0-5, 8.0-0, 8.0-3.

Karta **Znaczniki** umożliwia definiowanie znaczników na poziomie szablonu. Wszystkie problemy hostów połączonych z tym szablonem zostaną oznaczone wartościami wprowadzonymi tutaj.



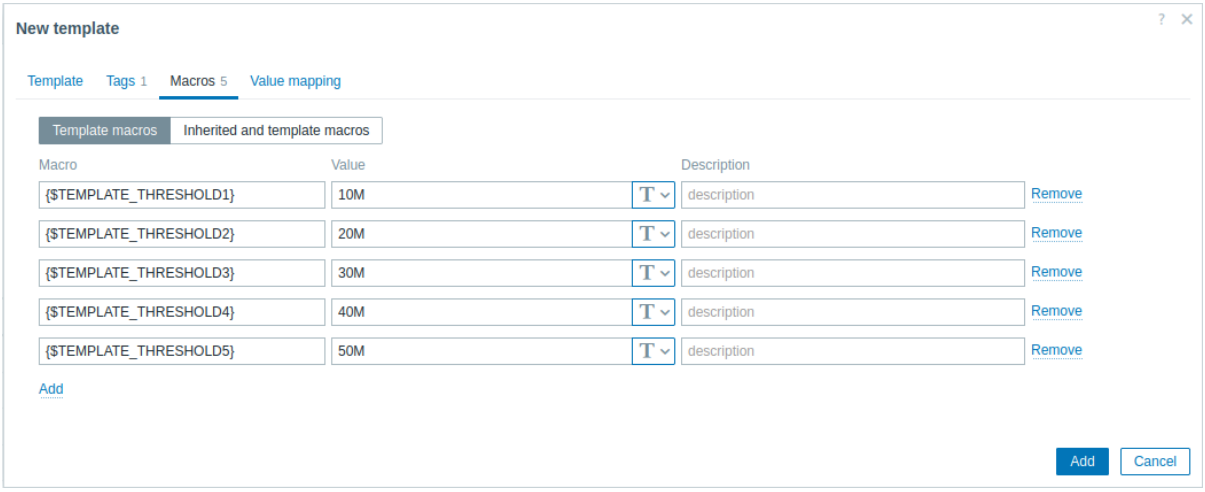
Jeśli wybierzesz opcję *Odziedziczone i znaczniki szablonu*, możesz również wyświetlać znaczniki z połączonych szablonów, które ten szablon odziedziczy.



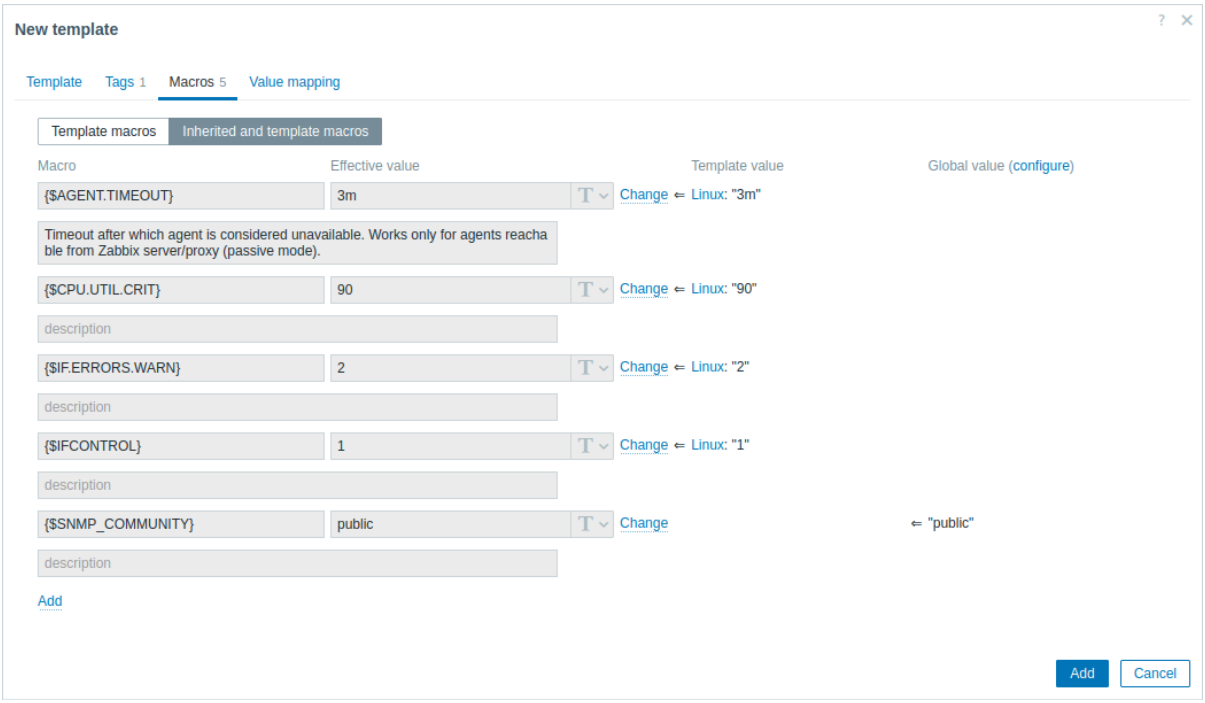
Dla wygody udostępniono odnośniki do odpowiednich szablonów.

W znacznikach obsługiwane są makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} i {HOST.ID}.

Karta **Makra** umożliwia definiowanie makr użytkownika na poziomie szablonu w postaci par nazwa-wartość. Należy pamiętać, że wartości makr mogą być przechowywane jako zwykły tekst, tekst poufny lub sekret Vault. Możliwe jest również dodanie opisu.



Jeśli wybierzesz opcję *Odziedziczone i makra szablonu*, możesz również wyświetlać makra z połączonych szablonów i makra globalne, które ten szablon odziedziczy, a także wartości, do których makra zostaną rozwinięte.



Dla wygody udostępniono odnośniki do odpowiednich szablonów oraz odnośnik do globalnej konfiguracji makr. Możliwe jest również edytowanie makra połączonego szablonu lub makra globalnego na poziomie szablonu, co skutecznie tworzy kopię makra w szablonie.

Karta **Mapowanie wartości** umożliwia skonfigurowanie przyjaznej dla użytkownika reprezentacji danych pozycji w mapowaniach wartości.

Przyciski:

Add	Dodaj szablon. Dodany szablon powinien pojawić się na liście.
Update	Zaktualizuj właściwości istniejącego szablonu.
Clone	Utwórz kolejny szablon na podstawie właściwości bieżącego szablonu. Obejmuje to encje (pozycje, wyzwalacze itd.) odziedziczone zarówno z połączonych szablonów, jak i bezpośrednio dołączone do bieżącego szablonu, ale wyklucza dostawcę i wersję bieżącego szablonu, aby sklonowany szablon można było odróżnić od oryginału.
Delete	Usuń szablon; encje szablonu (pozycje, wyzwalacze itd.) pozostaną przy połączonych hostach.
Delete and clear	Usuń szablon i wszystkie jego encje z połączonych hostów.
Cancel	Anuluj edytowanie właściwości szablonu.

Dodawanie pozycji, wyzwalaczy, wykresów

Attention:
Pozycje muszą najpierw zostać dodane do szablonu. Wyzwalaczy i wykresów nie można dodać bez odpowiedniej pozycji.

Możesz dodać istniejące pozycje do szablonu lub utworzyć nowe pozycje.

Aby dodać pozycje do szablonu:

1. Przejdź do *Zbieranie danych* > *Hosts* (lub *Szablony*).
2. Kliknij *Pozycje* w wierszu odpowiedniego hosta/szablonu.
3. Zaznacz pola wyboru przy pozycjach, które chcesz dodać do szablonu.
4. Kliknij *Kopiuj* poniżej listy pozycji.
5. Wybierz szablon (lub grupę szablonów), do którego mają zostać skopiowane pozycje, i kliknij *Kopiuj*.
 Wszystkie zaznaczone pozycje powinny zostać skopiowane do szablonu.

Dodawanie wyzwalaczy i wykresów odbywa się w podobny sposób (odpowiednio z listy wyzwalaczy i wykresów), przy czym należy pamiętać, że można je dodać tylko wtedy, gdy wcześniej dodano wymagane pozycje.

Dodawanie dashboardów

Aby dodać dashboardy do szablonu w *Data collection* > *Templates*, wykonaj następujące czynności:

1. Kliknij *Dashboards* w wierszu szablonu.
2. Skonfiguruj dashboard zgodnie z wytycznymi dotyczącymi konfigurowania dashboardów.

Attention:
Podczas konfigurowania widżetów na dashboardzie szablonu (zamiast na globalnym dashboardzie) parametry związane z hostem nie są dostępne, a niektóre parametry mają inną etykietę. Dzieje się tak, ponieważ dashboardy szablonów wyświetlają dane tylko z hosta, z którym szablon jest powiązany. Na przykład parametry *Host groups*, *Exclude host groups* i *Hosts* w widżecie *Problems* nie są dostępne, parametr *Host groups* w widżecie *Host availability* nie jest dostępny, a parametr *Show hosts in maintenance* został zmieniony na *Show data in maintenance* itd. Więcej informacji o dostępności parametrów w widżetach dashboardu szablonu można znaleźć w opisie poszczególnych parametrów dla każdego widżetu dashboardu.

Note:
Szczegółowe informacje o dostępie do dashboardów hosta, które są tworzone na podstawie dashboardów szablonu, znajdują się w sekcji host dashboards.

Konfigurowanie reguł wykrywania niskiego poziomu

Zobacz sekcję wykrywanie niskiego poziomu w podręczniku.

Dodawanie scenariuszy web

Aby dodać scenariusze web do szablonu w *Data collection* > *Templates*, wykonaj następujące czynności:

1. Kliknij *Web* w wierszu szablonu.
2. Skonfiguruj scenariusz web, postępując zgodnie ze standardową metodą konfigurowania scenariuszy web.

Konfigurowanie grupy szablonów

Przegląd

Grupy szablonów służą do logicznego grupowania szablonów oraz przypisywania do nich uprawnień użytkowników.

Każdy szablon musi mieć przypisaną co najmniej jedną grupę szablonów. Szablon może należeć do wielu grup szablonów, a każda grupa szablonów może zawierać wiele szablonów.

Należy pamiętać, że w Zabbix wszystkie uprawnienia są oparte na grupach: grupach użytkowników, grupach hostów oraz grupach szablonów. Dlatego nawet jeśli pojedynczy użytkownik potrzebuje dostępu do pojedynczego szablonu, przyznaje się go przez dodanie użytkownika do grupy użytkowników i przypisanie tej grupie użytkowników uprawnień do grupy szablonów zawierającej ten szablon.

Konfiguracja

Attention:
Tylko użytkownicy z uprawnieniami Super admin mogą tworzyć grupy szablonów.

W frontend Zabbix istnieją dwie opcje tworzenia grupy szablonów.

Opcja pierwsza:

1. Przejdź do *Data collection* > *Template groups*.
2. Kliknij *Create template group* w prawym górnym rogu ekranu.
3. Wprowadź nazwę grupy w formularzu.



Opcja druga: podczas konfigurowania szablonu wprowadź nieistniejącą nazwę grupy w polu *Template groups*.

Po utworzeniu grupy szablonów możesz kliknąć nazwę szablonu na liście w *Data collection* > *Template groups*, aby edytować nazwę grupy, sklonować grupę lub usunąć grupę.

Usunięcie grupy szablonów usuwa tylko grupę logiczną, a nie szablony w tej grupie. Nie można usunąć grupy szablonów, jeśli jest ona jedyną grupą dla dowolnego istniejącego szablonu.

Tworzenie podgrup szablonów

Podgrupa szablonu (lub zagnieżdżona grupa szablonów) jest elementem podrzędnym nadrzędnej grupy szablonów, która ją zawiera.

Podgrupę tworzy się przy użyciu ukośnika '/' w polu wprowadzania nazwy grupy, aby określić jej relację do grupy nadrzędnej lub grup nadrzędnych. Na przykład:

- wpisanie `Linux servers/Databases` tworzy podgrupę `Linux servers/Databases` w nadrzędnej grupie `Linux servers`.
- wpisanie `Linux servers/Databases/MySQL/Tokyo` tworzy odpowiednią podgrupę w zagnieżdżonych grupach nadrzędnych `Linux servers`, `Linux servers/Databases`, `Linux servers/Databases/MySQL`.

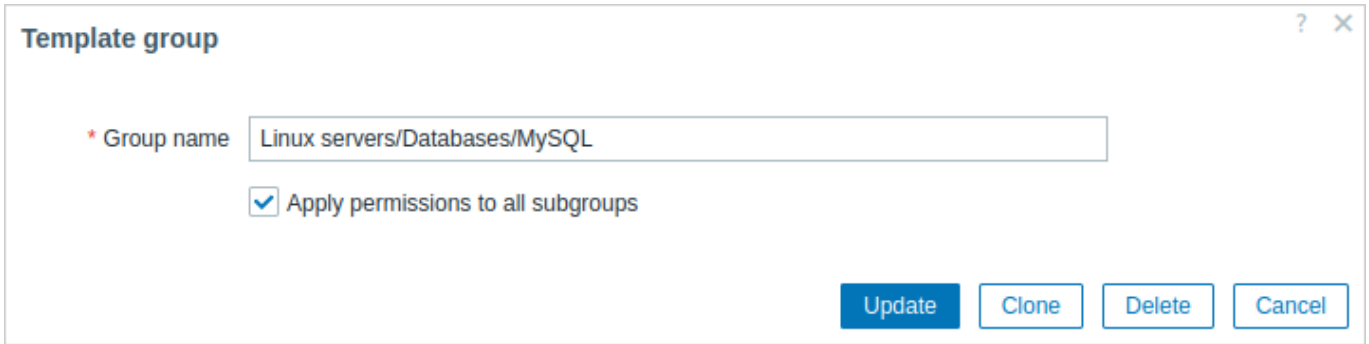
Podczas tworzenia podgrupy nie można używać ukośników na początku ani na końcu nazwy ani kilku ukośników z rzędu. Znak '/' nie obsługuje sekwencji ucieczki.

Nie jest wymagane tworzenie żadnych nadrzędnych grup szablonów przed utworzeniem podgrupy. Możesz zdecydować, czy zacząć od utworzenia podgrupy (na przykład `Linux servers/Databases`), czy dowolnej nadrzędnej grupy szablonów (w naszym przykładzie `Linux servers`). Jeśli zaczniesz od utworzenia podgrupy, nadrzędne grupy szablonów **nie** zostaną utworzone automatycznie.

Uprawnienia do grup szablonów

- Podczas tworzenia podgrupy w istniejącej nadrzędnej grupie szablonów (na przykład tworzenia `Linux servers/Databases`, gdy `Linux servers` już istnieje), uprawnienia grupy użytkowników do podgrupy są dziedziczone po grupie nadrzędnej.
- Podczas tworzenia nadrzędnej grupy szablonów dla istniejącej podgrupy (na przykład tworzenia `Linux servers`, gdy `Linux servers/Databases` już istnieje), nie są ustawiane żadne uprawnienia dla grupy nadrzędnej.

Podczas edycji dowolnej grupy szablonów można również ustawić dodatkową opcję *Zastosuj uprawnienia do wszystkich podgrup*.



Zaznaczenie tego pola wyboru i kliknięcie *Update* spowoduje zastosowanie tego samego poziomu uprawnień do wszystkich bieżących i przyszłych podgrup edytowanej grupy szablonów.

Jeśli więc jakimkolwiek grupom użytkowników nadano różne uprawnienia do podgrup edytowanej grupy szablonów, zaznaczenie pola wyboru nada wszystkim bieżącym i przyszłym podgrupom takie same uprawnienia użytkowników jak edytowanej grupie.

Należy pamiętać, że ta opcja nie jest zapisywana w bazie danych i zastąpi istniejące uprawnienia. Wszelkie zmiany wprowadzone za pomocą tej opcji można cofnąć wyłącznie ręcznie.

Łączenie/rozłączanie

Przegląd

Podłączanie to proces, w którym szablony są stosowane do hostów, natomiast odłączanie usuwa powiązanie z szablonem z hosta.

Łączenie szablonu

Aby połączyć szablon z hostem:

1. Przejdź do sekcji *Zbieranie danych* > *Hosty*.
2. Kliknij wymagany host.
3. Zacznij wpisywać nazwę szablonu w polu *Szablony*. Zostanie wyświetlona lista pasujących szablonów; przewiń ją, aby wybrać szablon. Możesz również kliknąć przycisk *Wybierz* obok pola i wybrać jeden lub kilka szablonów z listy w wyskakującym oknie.
4. Kliknij przycisk *Dodaj/Aktualizuj* w formularzu atrybutów hosta.

Host będzie teraz zawierał wszystkie encje szablonu. Obejmuje to pozycje, wyzwalacze, wykresy, reguły wykrywania niskiego poziomu, scenariusze webowe, a także pulpity nawigacyjne.

Attention:
Połączenie wielu szablonów z tym samym hostem nie powiedzie się, jeśli szablony te zawierają pozycje z tym samym kluczem pozycji. Ponieważ wyzwalacze i wykresy korzystają z pozycji, nie można ich również połączyć z jednym hostem z wielu szablonów, jeśli używają identycznych kluczy pozycji.

Gdy encje (pozycje, wyzwalacze itd.) zostaną dodane z szablonu:

- wcześniej istniejące identyczne encje na hoście zostaną zaktualizowane jako encje szablonu, a **wszelkie istniejące dostosowania encji na poziomie hosta zostaną utracone**;
- encje z szablonu zostaną dodane;
- wszelkie bezpośrednio połączone encje, które przed połączeniem szablonu istniały wyłącznie na hoście, pozostaną niezmienione.

Na listach wszystkie encje z szablonu będą teraz poprzedzone nazwą szablonu, wskazującą, że należą do konkretnego szablonu. Sama nazwa szablonu (wyświetlana na szaro) jest odnośnikiem umożliwiającym dostęp do listy tych encji na poziomie szablonu.

Note:
W przypadku niektórych pozycji, takich jak sprawdzanie zewnętrzne, sprawdzanie przez agenta HTTP, proste sprawdzanie, sprawdzanie SSH i sprawdzanie Telnet, interfejs hosta jest opcjonalny. Jeśli w momencie łączenia szablonu host nie ma zdefiniowanego interfejsu, te pozycje zostaną dodane bez interfejsu. Jeśli później dodasz interfejs hosta, nie zostanie on automatycznie przypisany do już istniejących pozycji. Aby jednocześnie przypisać nowo dodany interfejs hosta do wszystkich pozycji szablonu, odłącz szablon od hosta, a następnie połącz go ponownie. Aby zachować historię pozycji, użyj opcji *Odłącz*, a nie opcji *Odłącz i wyczyść*.

Jeśli jakaś encja nie jest poprzedzona nazwą szablonu, oznacza to, że istniała na hoście wcześniej i nie została dodana przez szablon.

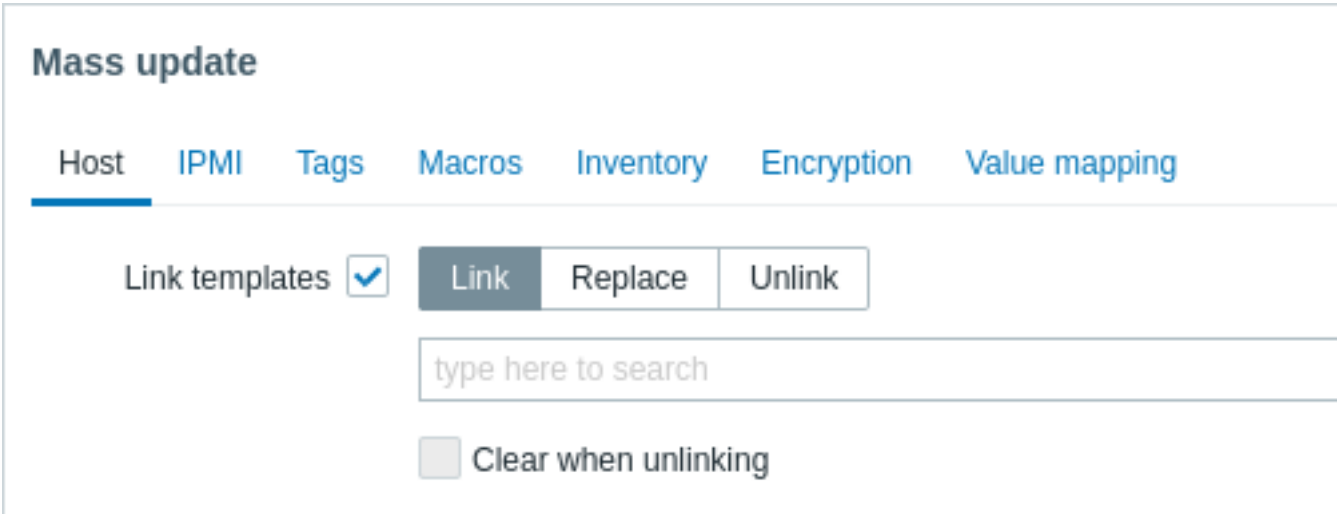
Kryteria unikalności encji

Podczas dodawania encji (pozycji, wyzwalaczy itp.) z szablonu ważne jest, aby wiedzieć, które z tych encji już istnieją na hoście i wymagają aktualizacji, a które encje się różnią. Kryteria unikalności służące do określenia, czy encje są takie same, czy różne, są następujące:

- dla pozycji - klucz pozycji;
- dla wyzwalaczy - nazwa wyzwalacza i wyrażenie;
- dla niestandardowych wykresów - nazwa wykresu i jego pozycje.

Łączenie szablonów z wieloma hostami

Aby zaktualizować powiązania szablonów wielu hostów, w sekcji *Zbieranie danych* > *Hosty* zaznacz wybrane hosty, zaznaczając ich pola wyboru, a następnie kliknij przycisk **Aktualizacja masowa** poniżej listy i wybierz opcję *Połącz szablony*:



Aby połączyć dodatkowe szablony, zacznij wpisywać nazwę szablonu w polu autouzupełniania, aż pojawi się lista rozwijana z pasującymi szablonami. Przewiń listę, aby wybrać szablon do połączenia.

Opcja *Zastąp* umożliwia połączenie nowego szablonu przy jednoczesnym odłączeniu wszystkich szablonów, które były wcześniej połączone z hostami. Opcja *Odłącz* umożliwia określenie szablonów, które mają zostać odłączone. Opcja *Wyczyść podczas odłączania* umożliwia nie tylko odłączenie wcześniej połączonych szablonów, ale także usunięcie wszystkich encji odziedziczonych z tych szablonów (pozycje, wyzwalacze itd.).

Note:
Zabbix oferuje obszerny zestaw predefiniowanych szablonów. Możesz używać ich jako punktu odniesienia, ale unikaj używania ich bez zmian w środowisku produkcyjnym, ponieważ mogą zawierać zbyt wiele pozycji i zbyt często odpytywać o dane. Jeśli chcesz ich używać, dostosuj je do swoich rzeczywistych potrzeb.

Edycja podlinkowanych encji

Jeśli spróbujesz edytować pozycja lub wyzwalacz, które zostały podlinkowane z szablon, możesz zauważyć, że wiele kluczowych opcji edycji jest wyłączonych. Ma to sens, ponieważ idea szablonów polega na tym, że elementy są edytowane centralnie na poziomie szablonu. Nadal jednak możesz na przykład włączać/wyłączać pozycja na poszczególnych hostach oraz ustawiać interwał aktualizacji, długość historii i niektóre inne parametry.

Jeśli chcesz w pełni edytować encję, musisz zrobić to na poziomie szablonu (skrót do poziomu szablonu jest wyświetlany w nazwie formularza), pamiętając, że te zmiany wpłyną na wszystkie hosty, które mają podlinkowany ten szablon.

Attention:
Wszelkie dostosowania encji wprowadzone na poziomie szablonu zastąpią wcześniejsze dostosowania encji na poziomie hosta.

Odłączanie szablonu

Aby odłączyć szablon od hosta, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Hosts*.
2. Kliknij wymagany host i znajdź pole *Templates*.
3. Kliknij *Unlink* lub *Unlink and clear* obok szablonu, aby odłączyć szablon.
4. Kliknij *Update* w formularzu atrybutów hosta.

Wybranie opcji *Unlink* spowoduje jedynie usunięcie powiązania z szablonem, przy jednoczesnym pozostawieniu wszystkich jego elementów na hoście. Dotyczy to pozycji, wyzwalaczy, wykresów, reguł wykrywania niskiego poziomu oraz scenariuszy webowych, ale nie obejmuje pulpitów. Należy pamiętać, że mapy wartości i tagi odziedziczone z powiązanych szablonów również zostaną usunięte.

Wybranie opcji *Unlink and clear* spowoduje usunięcie zarówno powiązania z szablonem, jak i wszystkich jego elementów (pozycji, wyzwalaczy itd.).

Zagnieżdżanie

Przegląd

Zagnieżdżanie to sposób, w jaki jeden szablon obejmuje jeden lub więcej innych szablonów.

Ponieważ sensowne jest rozdzielenie encji na osobne szablony dla różnych usług, aplikacji itp., możesz ostatecznie mieć całkiem sporo szablonów, z których wszystkie mogą wymagać podłączenia do wielu hostów. Aby uprościć ten obraz, możliwe jest połączenie kilku szablonów razem w jednym szablonie.

Korzyść z zagnieżdżania polega na tym, że do hosta trzeba podłączyć tylko jeden szablon, a host automatycznie odziedziczy wszystkie encje z szablonów, które są podłączone do tego jednego szablonu. Na przykład, jeśli podłączymy *T1* i *T2* do *T3*, uzupełnimy *T3* o wszystkie encje z *T1* i *T2*, ale nie odwrotnie. Jeśli podłączymy *T1* do *T2* i *T3*, uzupełnimy *T2* i *T3* o encje z *T1*.

Konfigurowanie zagnieżdżonych szablonów

Aby połączyć szablony, należy użyć istniejącego szablonu (lub utworzyć nowy), a następnie:

1. Otwórz formularz konfiguracji szablonu.
2. Znajdź pole *Templates*.
3. Kliknij *Select*, aby otworzyć okno podręczne *Templates*.
4. W oknie podręcznym wybierz wymagane szablony, a następnie kliknij *Select*, aby dodać szablony do listy.
5. Kliknij *Add* lub *Update* w formularzu konfiguracji szablonu.

W ten sposób wszystkie obiekty skonfigurowanego szablonu, a także wszystkie obiekty połączonych szablonów, będą teraz widoczne w konfiguracji szablonu. Dotyczy to pozycji, wyzwalaczy, wykresów, reguł wykrywania niskiego poziomu oraz scenariuszy web, ale nie obejmuje dashboardów. Jednak dashboardy z połączonych szablonów będą mimo to dziedziczone przez hosty.

Aby odłączyć dowolny z połączonych szablonów, kliknij *Unlink* lub *Unlink and clear* w formularzu konfiguracji szablonu, a następnie kliknij *Update*.

Opcja *Unlink* po prostu usunie powiązanie z połączonym szablonem, nie usuwając żadnych jego obiektów (pozycji, wyzwalaczy itp.).

Opcja *Unlink and clear* usunie zarówno powiązanie z połączonym szablonem, jak i wszystkie jego obiekty (pozycje, wyzwalacze itp.).

Masowa aktualizacja

Przegląd

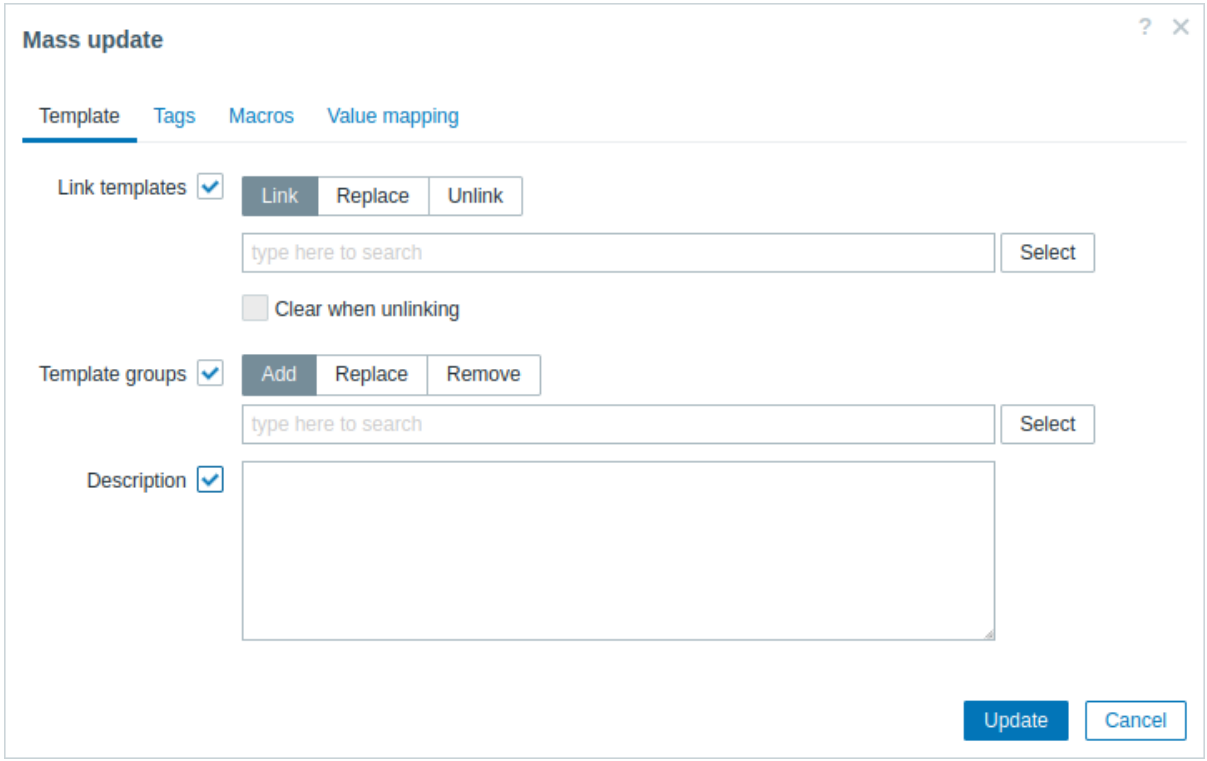
Czasami możesz chcieć zmienić jakiś atrybut dla wielu szablonów jednocześnie. Zamiast otwierać każdy pojedynczy szablon do edycji, możesz w tym celu użyć funkcji masowej aktualizacji.

Używanie masowej aktualizacji

Aby wykonać masową aktualizację niektórych szablonów, wykonaj następujące czynności:

1. Zaznacz pola wyboru przed szablonami, które chcesz zaktualizować na liście szablonów.
2. Kliknij *Mass update* poniżej listy.
3. Przejdź do karty z wymaganymi atrybutami (*Template*, *Tags*, *Macros* lub *Value mapping*).
4. Zaznacz pola wyboru dowolnych atrybutów do aktualizacji i wprowadź dla nich nową wartość.

Zakładka **Szablon** zawiera ogólne opcje masowej aktualizacji szablonów.



Po wybraniu odpowiedniego przycisku dla aktualizacji *Łączenie szablonów* dostępne są następujące opcje:

- *Połącz* - określ dodatkowe szablony, które mają zostać połączone;
- *Zastąp* - określ szablony, które mają zostać połączone, jednocześnie odłączając wszystkie wcześniej połączone szablony;
- *Odłącz* - określ szablony, które mają zostać odłączone.

Aby określić szablony do połączenia/odłączenia, zacznij wpisywać nazwę szablonu w polu autouzupełniania, aż pojawi się lista rozwijana z pasującymi szablonami. Przewiń listę, aby wybrać wymagane szablony.

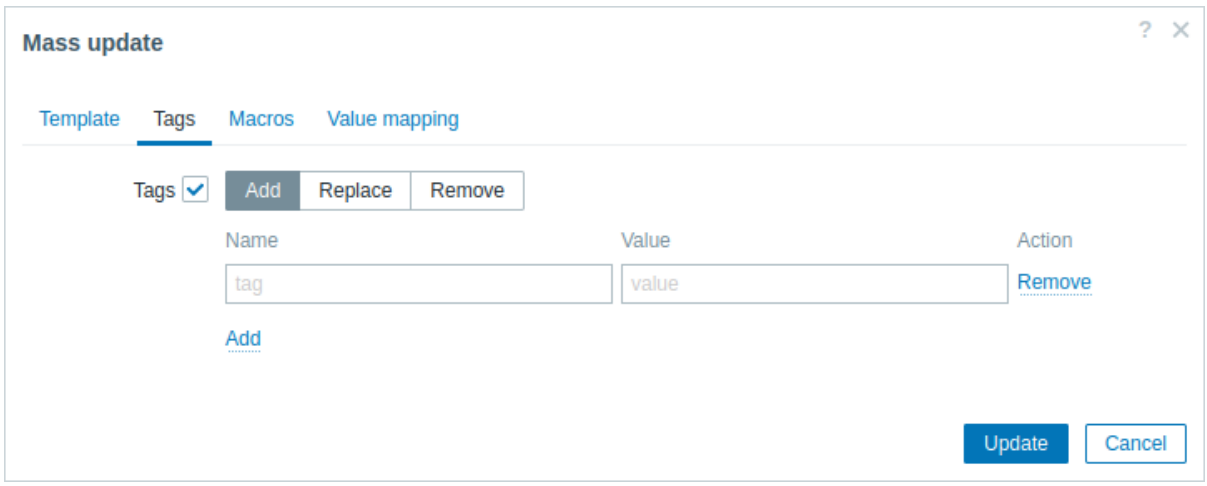
Opcja *Wyczyść podczas odłączania* umożliwia nie tylko odłączenie wcześniej połączonych szablonów, ale także usunięcie wszystkich encji odziedziczonych z tych szablonów (pozycji, wyzwalaczy, wykresów itp.).

Po wybraniu odpowiedniego przycisku dla aktualizacji *Grupy szablonów* dostępne są następujące opcje:

- *Dodaj* - określ dodatkowe grupy szablonów spośród istniejących lub wprowadź całkowicie nowe grupy szablonów dla szablonów;
- *Zastąp* - usuń szablon ze wszystkich istniejących grup szablonów i zastąp je grupą lub grupami określonymi w tym polu (istniejącymi lub nowymi grupami szablonów);
- *Usuń* - usuń określone grupy szablonów z szablonów.

Te pola obsługują autouzupełnianie - rozpoczęcie wpisywania tekstu powoduje wyświetlenie listy rozwijanej z pasującymi grupami szablonów. Jeśli grupa szablonów jest nowa, również pojawi się na liście rozwijanej i będzie oznaczona ciągiem *(nowa)*. Przewiń listę, aby wybrać grupę.

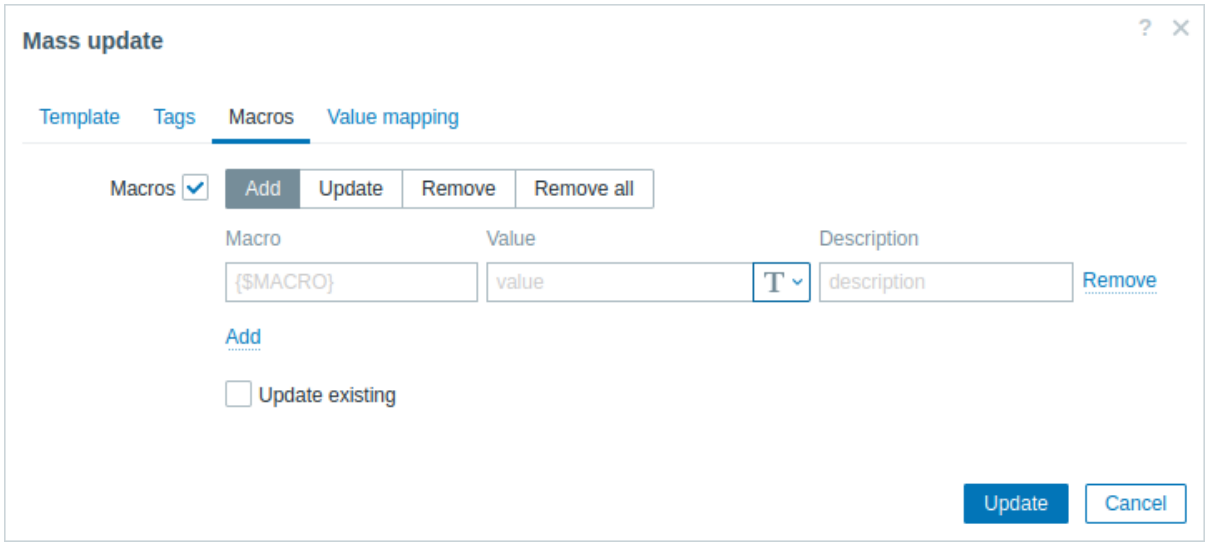
Zakładka **Tagi** umożliwia masową aktualizację tagów na poziomie szablonu.



Makra użytkownika, makra {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT}

oraz {HOST.ID} są obsługiwane w tagach. Należy pamiętać, że tagi o tej samej nazwie, ale o różnych wartościach, nie są uznawane za „duplikaty” i mogą zostać dodane do tego samego szablonu.

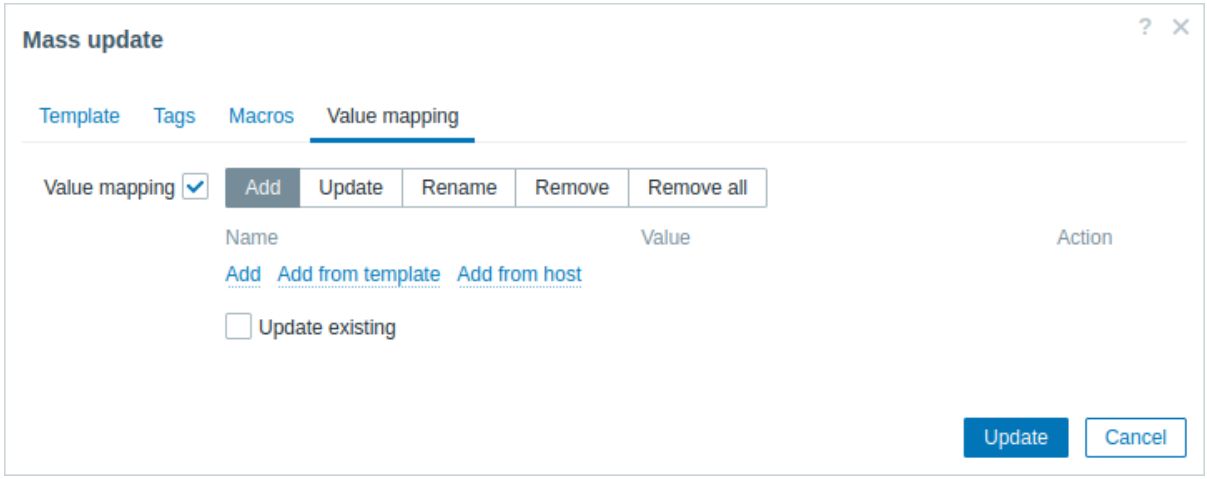
Zakładka **Makra** umożliwia masową aktualizację makr na poziomie szablonu.



Po wybraniu odpowiedniego przycisku dostępne są następujące opcje aktualizacji makr:

- *Dodaj* - określenie dodatkowych makr użytkownika dla szablonów. Jeśli pole wyboru *Aktualizuj istniejące* jest zaznaczone, wartość, typ i opis określonej nazwy makra zostaną zaktualizowane. Jeśli pole nie jest zaznaczone, a makro o tej nazwie już istnieje w szablonie lub szablonach, nie zostanie zaktualizowane.
- *Aktualizuj* - zastąpienie wartości, typów i opisów makr określonych na tej liście. Jeśli pole wyboru *Dodaj brakujące* jest zaznaczone, makro, które wcześniej nie istniało w szablonie, zostanie dodane jako nowe makro. Jeśli pole nie jest zaznaczone, zostaną zaktualizowane tylko makra, które już istnieją w szablonie.
- *Usuń* - usunięcie określonych makr z szablonów. Jeśli pole wyboru *Z wyjątkiem zaznaczonych* jest zaznaczone, zostaną usunięte wszystkie makra z wyjątkiem tych określonych na liście. Jeśli pole nie jest zaznaczone, zostaną usunięte tylko makra określone na liście.
- *Usuń wszystkie* - usunięcie wszystkich makr użytkownika z szablonów. Jeśli pole wyboru *Potwierdzam usunięcie wszystkich makr* nie jest zaznaczone, otworzy się nowe okno podręczne z prośbą o potwierdzenie usunięcia wszystkich makr.

Karta **Mapowanie wartości** umożliwia masową aktualizację mapowań wartości.



Dostępne są przyciski z następującymi opcjami aktualizacji map wartości:

- *Dodaj* - dodaje mapy wartości do szablonów. Jeśli zaznaczysz opcję *Aktualizuj istniejące*, wszystkie właściwości mapy wartości o tej nazwie zostaną zaktualizowane. W przeciwnym razie, jeśli mapa wartości o tej nazwie już istnieje, nie zostanie zaktualizowana.
- *Aktualizuj* - aktualizuje istniejące mapy wartości. Jeśli zaznaczysz opcję *Dodaj brakujące*, mapa wartości, która wcześniej nie istniała w szablonie, zostanie dodana jako nowa mapa wartości. W przeciwnym razie zostaną zaktualizowane tylko mapy wartości, które już istnieją w szablonie.
- *Zmień nazwę* - nadaje nową nazwę istniejącej mapie wartości.
- *Usuń* - usuwa określone mapy wartości z szablonów. Jeśli zaznaczysz opcję *Z wyjątkiem wybranych*, wszystkie mapy wartości zostaną usunięte **z wyjątkiem** tych, które zostały określone.

- *Usuń wszystkie* - usuwa wszystkie mapy wartości z szablonów. Jeśli pole wyboru *Potwierdzam usunięcie wszystkich map wartości* nie jest zaznaczone, otworzy się nowe okno podręczne z prośbą o potwierdzenie usunięcia.

Opcje *Dodaj z szablonu* i *Dodaj z hosta* są dostępne podczas dodawania/aktualizowania mapowania wartości. Umożliwiają one wybranie map wartości odpowiednio z szablonu lub hosta.

Po zakończeniu wszystkich wymaganych zmian kliknij *Update*. Atrybuty zostaną odpowiednio zaktualizowane dla wszystkich wybranych szablonów.

Hosty i grupy hostów

Czym jest „host”?

W Zabbix „host” oznacza dowolne urządzenie fizyczne lub wirtualne, aplikację, usługę albo dowolny inny logicznie powiązany zbiór monitorowanych parametrów.

Tworzenie hostów jest jednym z pierwszych zadań związanych z monitorowaniem w Zabbix. Na przykład, jeśli chcesz monitorować niektóre parametry na serwerze „x”, musisz najpierw utworzyć host o nazwie, na przykład, „Serwer X”, a następnie możesz dodać do niego pozycje monitorowania.

Hosty są organizowane w grupy hostów.

Przejdź do:

- Kreator hosta — aby skorzystać z prowadzonego krok po kroku interfejsu do tworzenia i konfigurowania nowego lub istniejącego hosta.
- Konfigurowanie hosta — aby skorzystać z klasycznego podejścia do tworzenia i konfigurowania hosta.

Kreator hosta

Omówienie

Host Wizard to prowadzony krok po kroku interfejs służący do konfigurowania monitorowanego celu (urządzenia, aplikacji, usługi itp.) w Zabbix. Prowadzi on przez następujące etapy:

- Wybór szablonu
- Utworzenie lub wybór hosta
- Instalacja agent lub agent 2 Zabbix
- Dodanie interfejsu hosta
- Zastosowanie dodatkowej konfiguracji do monitorowanego celu lub hosta Zabbix (jeśli jest to wymagane przez szablon)

Aby uzyskać dostęp do Host Wizard w frontend Zabbix, wykonaj następujące czynności:

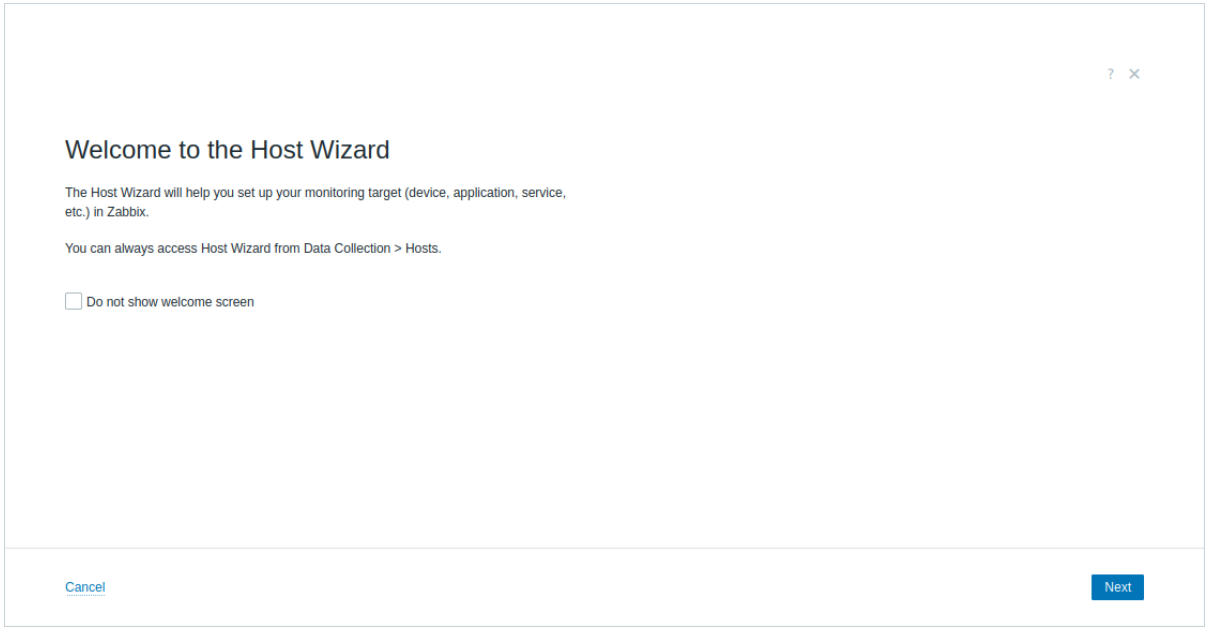
1. Przejdź do: *Data collection* > *Hosts*.
2. Kliknij *Host Wizard* w prawym górnym rogu ekranu.
3. Postępuj zgodnie z instrukcjami w kreatorze.

Możesz również edytować istniejącego hosta, klikając jego nazwę (lub ikonę trzech kropek obok niej) w różnych sekcjach frontend, aby uzyskać dostęp do Host Wizard z menu hosta. Należy pamiętać, że niektóre kroki są pomijane podczas edycji istniejącego hosta, jeśli zawiera on już konfigurację wymaganą przez wybrany szablon.

Alternatywnie możesz skorzystać z klasycznego sposobu tworzenia i konfigurowania hosta.

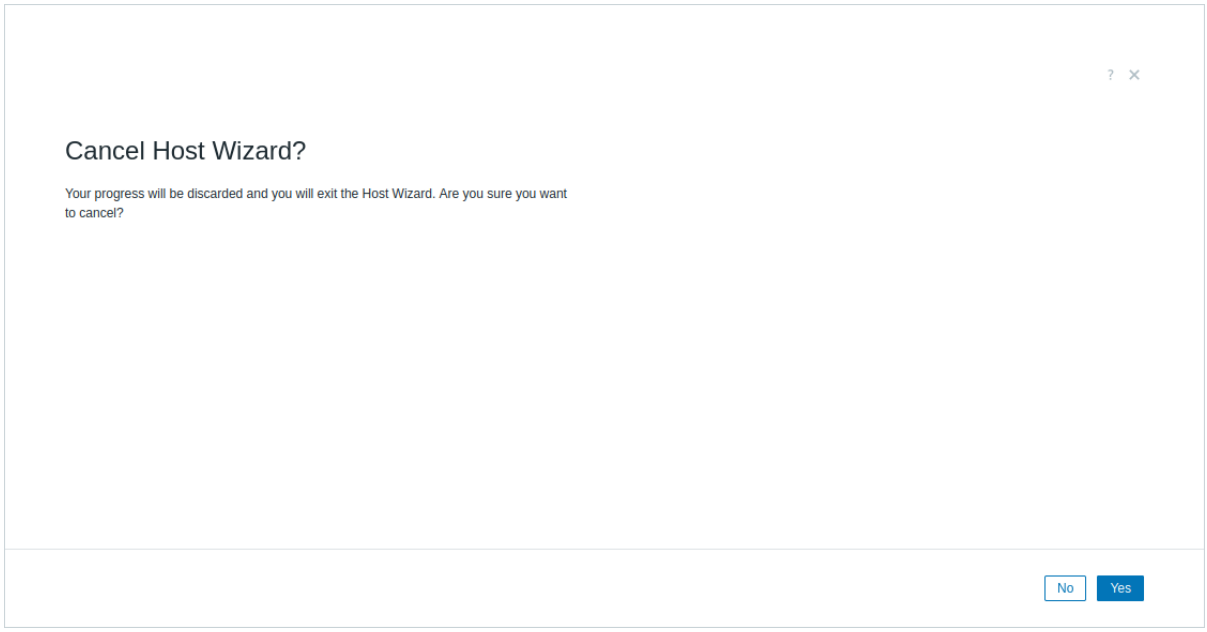
Witaj w kreatorze hosta

Po uruchomieniu Kreatora hosta wyświetlany jest ekran powitalny:



Aby pominąć ten ekran w przyszłych sesjach, możesz zaznaczyć pole wyboru *Nie pokazuj ekranu powitalnego* i kliknąć *Dalej*.

Jeśli rozpoczniesz konfigurowanie ustawień w kolejnych krokach Kreatora hosta i spróbujesz zamknąć kreator, klikając ikonę zamknięcia lub naciskając Esc, zostanie wyświetlony ekran potwierdzenia:



Kliknięcie *Tak* spowoduje wyjście z Kreatora hosta bez zapisywania postępu. Kliknięcie *Nie*, Esc lub ikony zamknięcia spowoduje powrót do ostatniego kroku.

Wybierz szablon

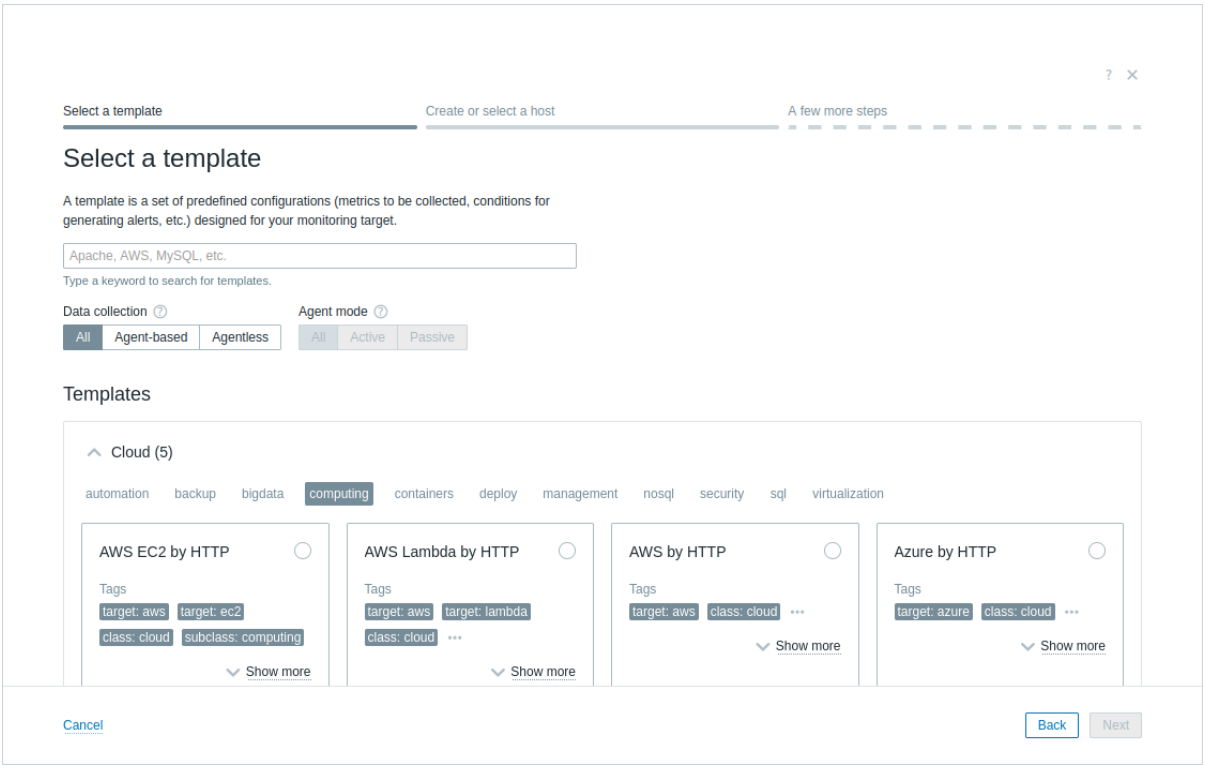
Pierwszym krokiem konfiguracji monitorowanego celu jest wybranie szablonu, czyli zestawu wstępnie zdefiniowanych konfiguracji (metryk do zebrania, warunków generowania alertów itp.) przeznaczonych dla monitorowanego celu.

Dostarczane wraz z Zabbixem szablony out-of-the-box oferują różnorodne wstępnie zdefiniowane konfiguracje monitorowania systemów operacyjnych, aplikacji, baz danych, urządzeń sieciowych, usług i innych elementów.

Na tym etapie możesz:

- Przeglądać szablony według klasy (Cloud, Database, Network itd.) i podklasy (np. automation, backup), które są oparte na znacznikach szablonów.
- Wyszukiwać szablony według słów kluczowych w nazwie szablonu lub nazwach i wartościach jego znaczników.
- Filtrować szablony według metody zbierania danych (opartej na agencie lub bez agenta; szablony oparte na agencie zawierają co najmniej jedną pozycję Zabbix agent).
- Filtrować szablony według trybu agenta (aktywny lub pasywny; szablony aktywne zawierają co najmniej jedną pozycję Zabbix agent (active), natomiast szablony pasywne zawierają co najmniej jedną pozycję pasywną; szczegółowe informacje znajdziesz w sekcji Pasywne i aktywne sprawdzanie przez agenta).

- Podczas korzystania z Host Wizard w celu skonfigurowania istniejącego hosta możesz również filtrować szablony, które są już powiązane z hostem.



Attention:
Podczas wybierania szablonu może zostać wyświetlony następujący komunikat: *Niektóre szablony (n) są niezgodne z Host Wizard. Dowiedz się, jak je zaktualizować. Szablony niestandardowe nie są obsługiwane.* Oznacza to, że niektóre szablony out-of-the-box nie są jeszcze zgodne z Host Wizard i wymagają aktualizacji; zapoznaj się z instrukcjami dotyczącymi aktualizacji szablonów.

Wybrany szablon określi kolejne kroki w Host Wizard. Na przykład jeśli wybierzesz szablon *MySQL by Zabbix agent*, który używa Zabbix agent do zbierania danych, kreator przeprowadzi Cię przez proces instalacji agenta.

Utwórz lub wybierz host

Szablon musi być powiązany z hostem, czyli jednostką w systemie Zabbix reprezentującą monitorowany cel. Po powiązaniu host otrzymuje wszystkie encje szablonu, takie jak pozycje (metryki do zebrania) i wyzwalacze (warunki generowania alertów), wymagane do monitorowania.

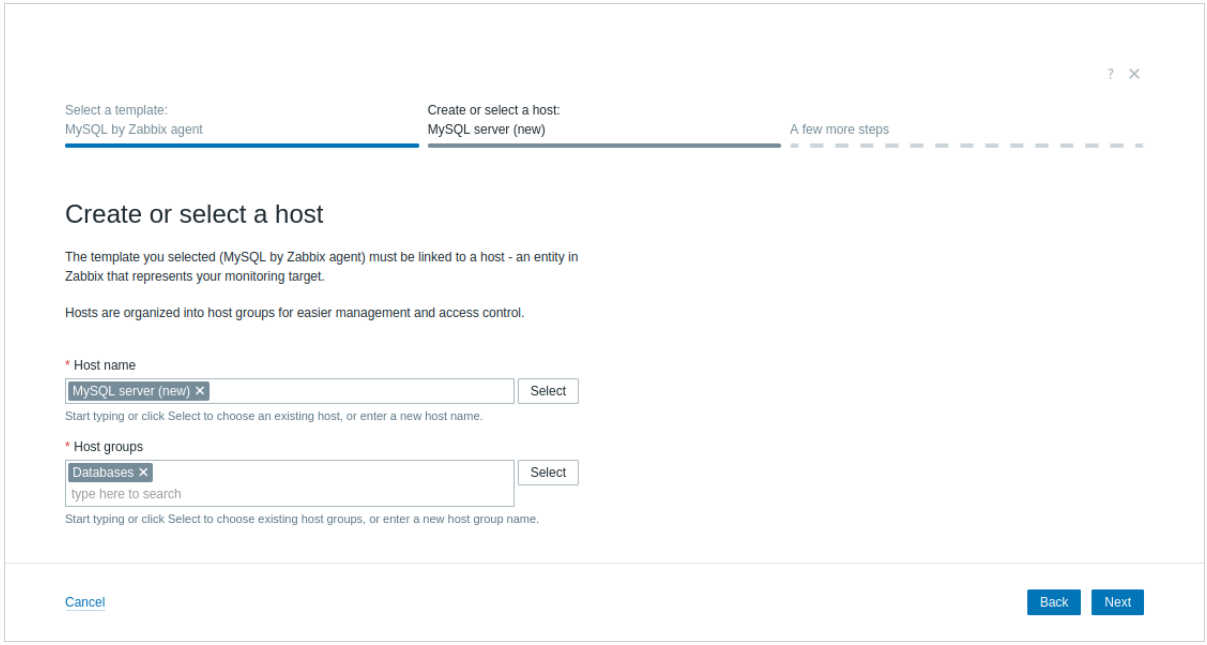
Każdy host musi również należeć do co najmniej jednej grupy hostów, która służy do organizowania hostów i przypisywania użytkownikom uprawnień do nich.

Na tym etapie możesz:

- Utworzyć nowy host i grupę hostów, wprowadzając nazwy obu elementów.
- Utworzyć nowy host, wprowadzając jego nazwę, i wybrać istniejącą grupę hostów, do której zostanie przypisany nowy host.
- Wybrać istniejący host; możesz również przypisać ten host do dodatkowych grup hostów bez usuwania go z żadnej z obecnych grup.

Note:
Wybranie istniejącego hosta nie usuwa ani nie zastępuje jego bieżącej konfiguracji, chyba że zostanie to wyraźnie określone w kolejnych krokach. Nie można wybrać wykrytych hostów.

Możesz na przykład utworzyć nowy host o nazwie *MySQL server*, reprezentujący lokalnie zainstalowany serwer MySQL, który chcesz monitorować. Wcześniej wybrany szablon *MySQL by Zabbix agent* zostanie powiązany z tym hostem. Dodatkowo możesz wybrać istniejącą grupę hostów *Databases* (lub utworzyć nową, np. *MySQL servers*), aby uporządkować host wraz z innymi hostami związanymi z bazami danych i później uprościć zarządzanie uprawnieniami.



Instalacja Zabbix agent

Szablony oparte na agencie (takie jak *MySQL by Zabbix agent*) wymagają zainstalowania Zabbix agent lub agent 2 — procesu wdrażanego na monitorowanym obiekcie w celu aktywnego monitorowania lokalnych zasobów i aplikacji (jeśli wymaga tego szablon).

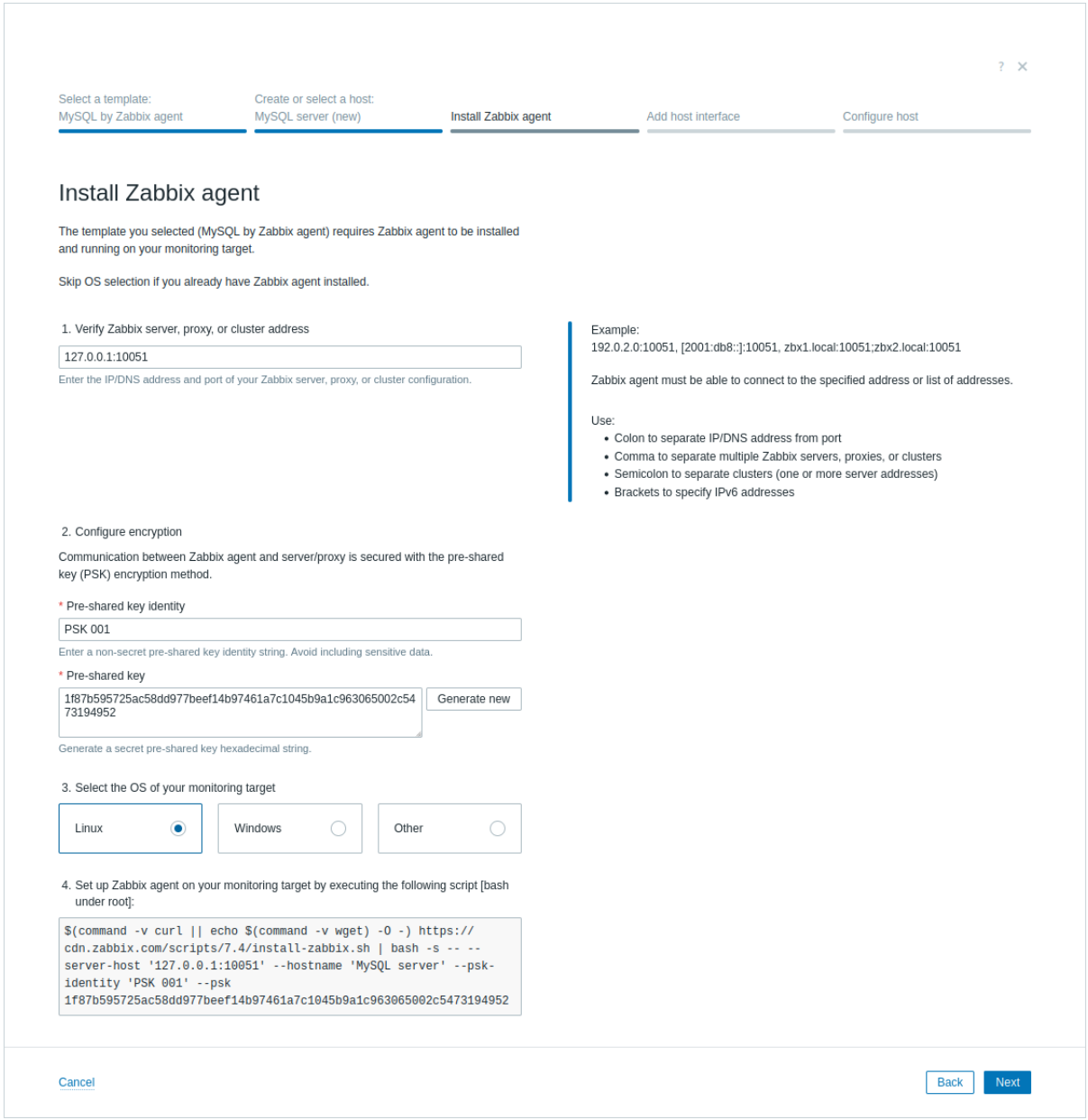
W tym kroku:

- Zweryfikuj adres serwera Zabbix, proxy lub klastra (np. 192.0.2.1:10051). Ta wartość zostanie użyta do skonfigurowania parametrów Server i ServerActive dla Zabbix agent.
- Wprowadź nową, unikatową i niebędącą sekretem tożsamość klucza wstępnie współdzielonego (np. *PSK 001* lub *mysql-agent-psk1*) i wygeneruj klucz wstępnie współdzielony. Te wartości zostaną użyte do skonfigurowania szyfrowania za pomocą klucza wstępnie współdzielonego (PSK) dla hosta Zabbix (np. *serwera MySQL*) i Zabbix agent.

Attention:
Podczas edytowania istniejącego hosta konfiguracja PSK zastąpi wszystkie istniejące ustawienia szyfrowania hosta.

- Wybierz system operacyjny maszyny, na której znajduje się monitorowany obiekt.
- Zainstaluj Zabbix agent w tym systemie, uruchamiając na nim dostarczony skrypt lub postępując zgodnie z instrukcjami instalacji.

Na przykład możesz ustawić *Tożsamość klucza wstępnie współdzielonego* na *PSK 001* i wygenerować nowy *Klucz wstępnie współdzielony*. Następnie możesz wybrać *Linux* jako system operacyjny i uruchomić dostarczony skrypt w tym systemie. Po zainstalowaniu Zabbix agent wróć do Kreatora hosta.

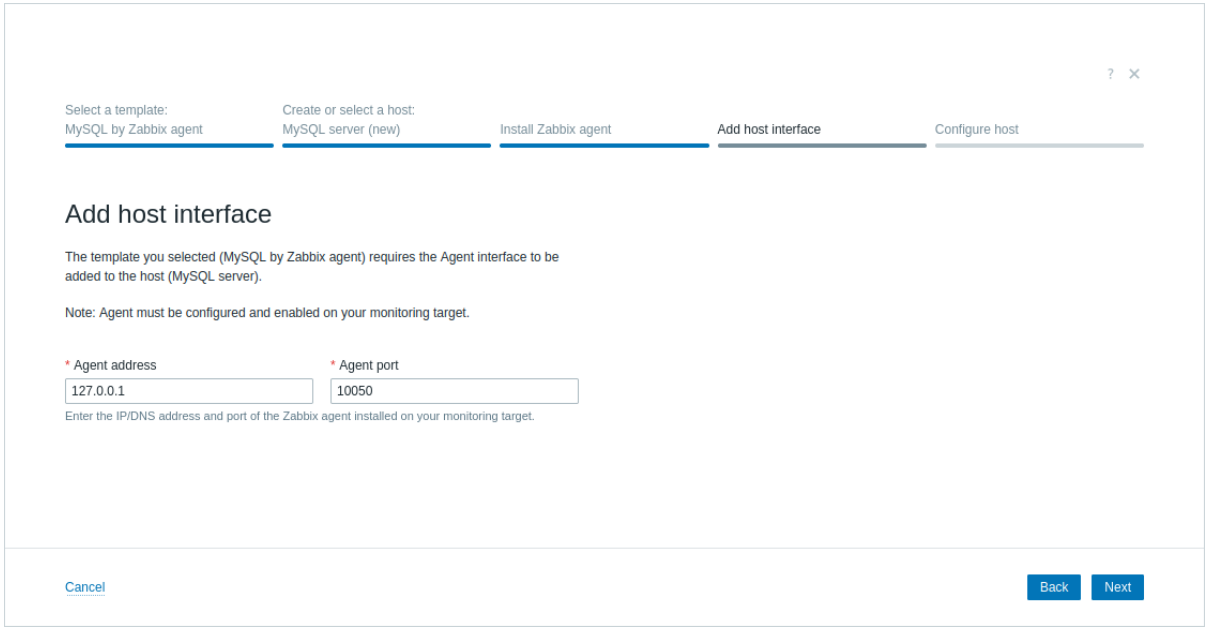


Dodaj interfejs hosta

Interfejs hosta definiuje, w jaki sposób serwer Zabbix łączy się z monitorowanym celem przez sieć. Określa parametry połączenia, takie jak adres IP, nazwa DNS, port oraz typ interfejsu (Agent, SNMP, JMX lub IPMI), w zależności od metody zbierania danych wymaganej przez wybrany szablon.

W tym kroku wprowadź szczegóły interfejsu hosta wymagane przez wybrany szablon.

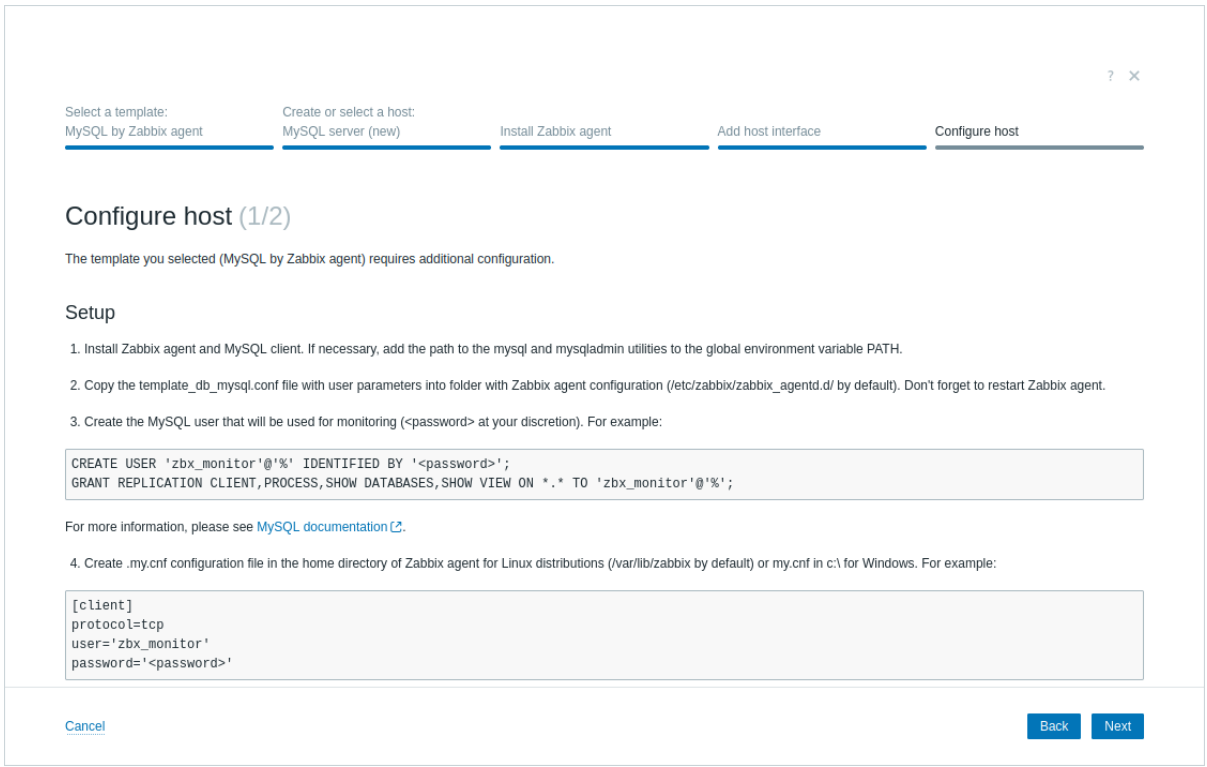
Na przykład możesz użyć domyślnego adresu agenta (`127.0.0.1`) i portu (10050), jeśli serwer Zabbix, agent Zabbix i serwer MySQL działają lokalnie na tej samej maszynie.



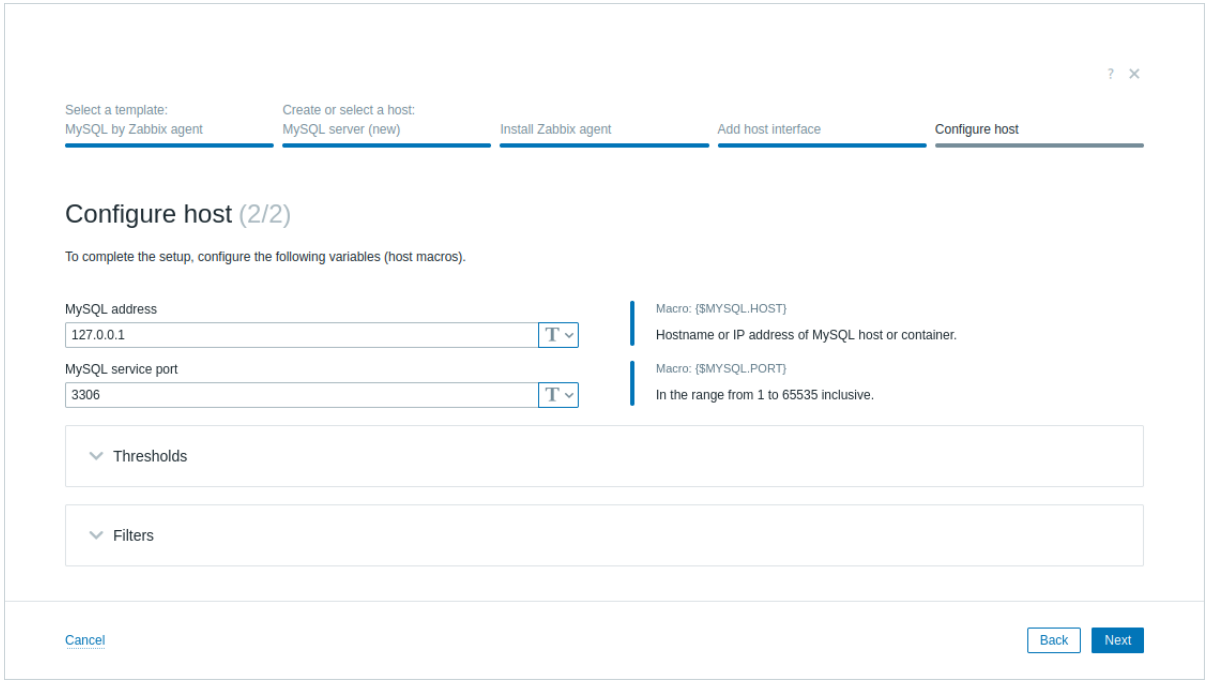
Konfigurowanie hosta

Niektóre szablony wymagają dodatkowej konfiguracji przed utworzeniem hosta. Może ona obejmować:

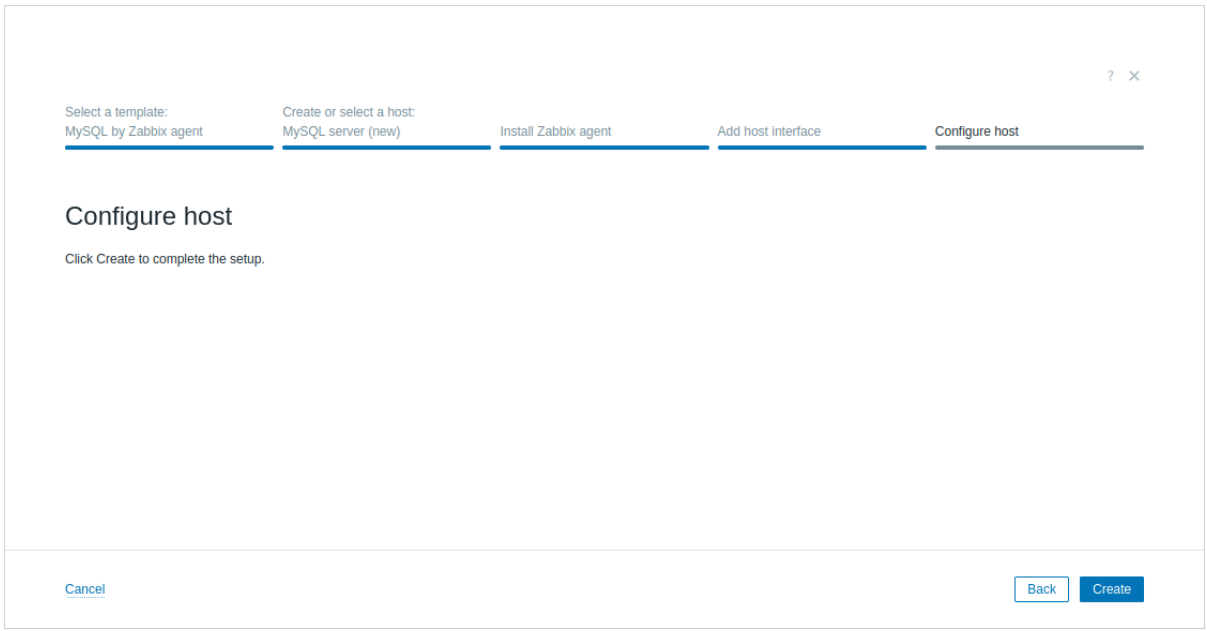
- Ręczną konfigurację monitorowanego celu (np. włączenie określonych usług, utworzenie użytkowników usług lub przyznanie uprawnień):



- Zdefiniowanie makr hosta - zmiennych sterujących działaniem pozycji, ustawieniami połączenia, danymi uwierzytelniającymi i innymi parametrami:



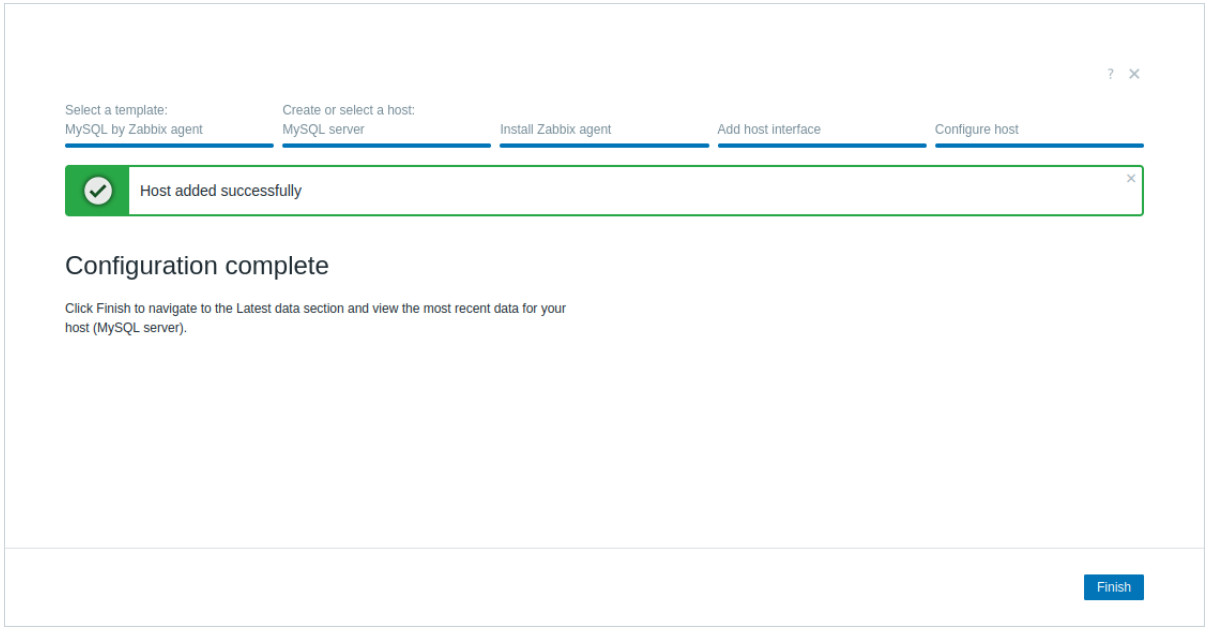
Po wykonaniu wszystkich wymaganych czynności konfiguracyjnych kliknij *Utwórz*, aby dodać hosta do Zabbix (lub *Aktualizuj*, aby zaktualizować konfigurację istniejącego hosta):



Konfiguracja zakończona

W tym momencie Zabbix monitoruje już wskazany cel.

W przypadku nowych hostów kliknij *Zakończ*, aby przejść do sekcji Najnowsze dane i wyświetlić najnowsze dane dla swojego hosta. W przypadku istniejących hostów kliknij *Zakończ*, aby zamknąć Kreator hosta i wrócić do ekranu, z którego został otwarty.



Konfigurowanie hosta

Przegląd

Aby skonfigurować host w frontend Zabbix, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Hosts* lub *Monitoring* > *Hosts*.
2. Kliknij *Create host* w prawym górnym rogu ekranu (lub nazwę hosta, aby edytować istniejący host).
3. Wprowadź parametry hosta w formularzu.

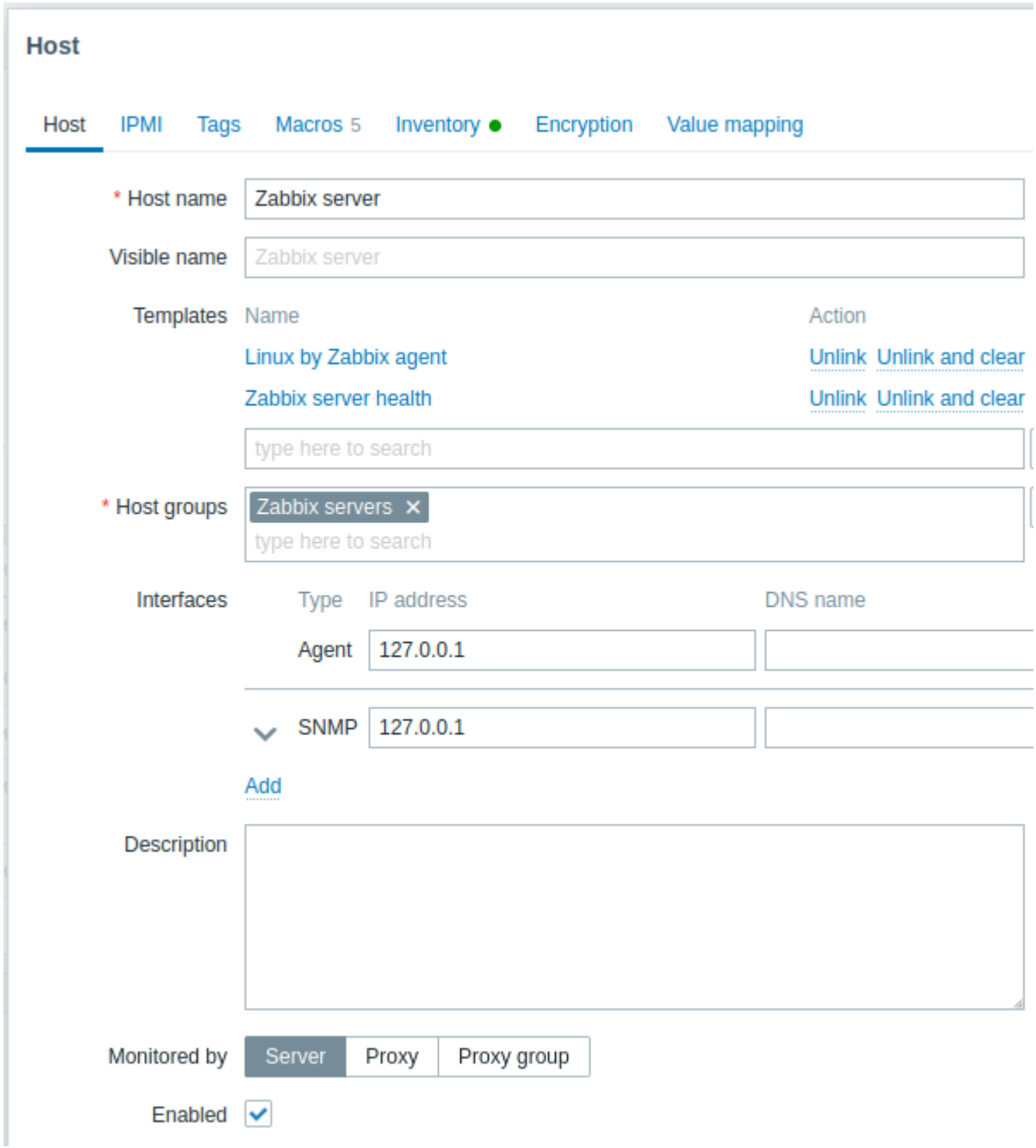
Możesz także użyć przycisku *Clone* w formularzu konfiguracji istniejącego hosta, aby utworzyć nowy host. Ten host będzie miał wszystkie właściwości istniejącego hosta, w tym powiązane szablony, elementy (pozycje, wyzwalacze itp.) z tych szablonów, a także elementy bezpośrednio przypisane do istniejącego hosta.

Należy pamiętać, że po sklonowaniu host zachowa wszystkie elementy szablonu dokładnie w takiej postaci, w jakiej występują pierwotnie w szablonie. Wszelkie zmiany w tych elementach wprowadzone na poziomie istniejącego hosta (takie jak zmieniony interwał pozycji, zmodyfikowane wyrażenie regularne lub dodane prototypy do reguły niskopoziomowego wykrywania) nie zostaną sklonowane do nowego hosta; zamiast tego będą takie jak w szablonie.

Alternatywnie możesz użyć Host Wizard, aby skonfigurować host za pomocą prowadzonego interfejsu krok po kroku.

Konfiguracja

Karta **Host** zawiera ogólne atrybuty hosta:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

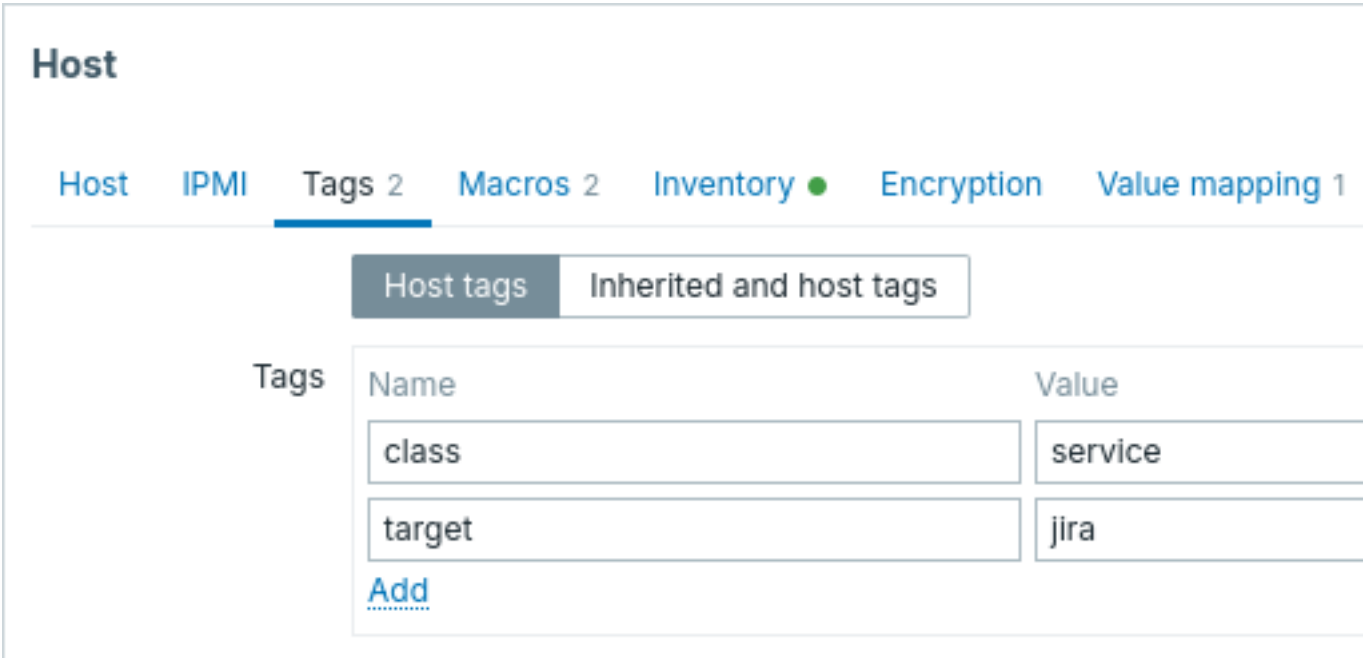
Parametr	Opis
Nazwa hosta	Wprowadź unikatową nazwę hosta. Dozwolone są znaki alfanumeryczne, spacje, kropki, myślniki i podkreślenia. Spacje na początku i na końcu są niedozwolone. *Uwaga:* Jeśli na konfigurowanym hoście działa agent Zabbix, parametr *Hostname* w pliku konfiguracyjnym agenta musi mieć taką samą wartość jak nazwa hosta wprowadzona tutaj. Nazwa określona w tym parametrze jest potrzebna do przetwarzania aktywnych kontroli.
Nazwa widoczna	Wprowadź unikatową nazwę widoczną hosta. Jeśli ustawisz tę nazwę, będzie ona wyświetlana na listach, mapach itd. zamiast technicznej nazwy hosta. Ten atrybut obsługuje kodowanie UTF-8.

Parametr	Opis
Szablony	Połącz szablony z hostem. Wszystkie encje (pozycje, wyzwalacze itd.) zostaną odziedziczone z szablonu. Aby połączyć nowy szablon, zacznij wpisywać jego nazwę w polu tekstowym. Zostanie wyświetlona lista pasujących szablonów; przewiń ją i wybierz szablon. Możesz również kliknąć *Wybierz* obok pola i wybrać szablony z listy w wyskakującym oknie. Wszystkie wybrane szablony zostaną połączone z hostem po zapisaniu lub zaktualizowaniu formularza konfiguracji hosta. Aby odłączyć szablon, użyj jednej z dwóch opcji w bloku *Połączone szablony*: *Odłącz* - odłącza szablon, ale zachowuje jego encje (pozycje, wyzwalacze itd.); *Odłącz i wyczyść* - odłącza szablon i usuwa wszystkie jego encje (pozycje, wyzwalacze itd.). Wyświetlane nazwy szablonów są klikalnymi odnośnikami prowadzącymi do formularza konfiguracji szablonu.
Grupy hostów	Wybierz grupy hostów, do których należy host. Host musi należeć do co najmniej jednej grupy hostów. Nową grupę można utworzyć i połączyć z hostem, dodając nieistniejącą nazwę grupy.
Interfejsy	Host może obsługiwać kilka typów interfejsów: *Agent*, *SNMP*, *JMX* i *IPMI*. Domyślnie nie są zdefiniowane żadne interfejsy. Aby dodać nowy interfejs, kliknij *Dodaj* w bloku *Interfejsy*, wybierz typ interfejsu i wprowadź informacje *IP/DNS*, *Połącz z* oraz *Port*. *Uwaga:* Interfejsów używanych przez dowolne pozycje nie można usunąć, a odnośnik *Usuń* jest dla nich nieaktywny. Wartość „IP” lub „DNS” z interfejsu SNMP jest również używana dla pułapek SNMP. Podczas dopasowywania używana jest tylko wybrana wartość „IP” lub „DNS” z interfejsu hosta. Dodatkowe informacje dotyczące konfigurowania interfejsu SNMP (v1, v2 i v3) zawiera sekcja Konfigurowanie monitorowania SNMP.
Adres IP	Adres IP hosta (opcjonalnie).
Nazwa DNS	Nazwa DNS hosta (opcjonalnie).
Połącz z	Kliknięcie odpowiedniego przycisku określi, czego serwer Zabbix użyje do pobierania danych z agentów: **IP** - połączenie z adresem IP hosta (zalecane) **DNS** - połączenie z nazwą DNS hosta
Port	Numer portu TCP/UDP. Wartości domyślne to: 10050 dla agenta Zabbix, 161 dla agenta SNMP, 12345 dla JMX i 623 dla IPMI.
Domyślny	Zaznacz przycisk radiowy, aby ustawić interfejs domyślny.
Opis	Wprowadź opis hosta.
Monitorowany przez	Wybierz, czy host jest monitorowany przez: **Serwer** - host jest monitorowany przez serwer Zabbix; **Proxy** - host jest monitorowany przez niezależny proxy; **Grupę proxy** - host jest monitorowany przez grupę proxy.
Proxy	Wyświetlana jest nazwa przypisanego proxy (tylko jeśli serwer Zabbix przypisał proxy z wybranej grupy proxy). To pole jest wyświetlane tylko wtedy, gdy host jest monitorowany przez grupę proxy.
Włączony	Po zaznaczeniu pola wyboru host jest włączony i gotowy do monitorowania. Po usunięciu zaznaczenia host jest wyłączony i nie jest monitorowany: W przypadku pasywnych żądań danych inicjowanych przez serwer Zabbix/proxy (na przykład agenta Zabbix, agenta SNMP, kontroli prostych) monitorowanie zostanie wyłączone po synchronizacji konfiguracji. Wyzwalacze i akcje powiązane z hostem również zostaną wyłączone dopiero po ponownym załadowaniu konfiguracji. W przypadku aktywnych kontroli agenta Zabbix monitorowanie zostanie zatrzymane w czasie (około 5 sekund), w którym agent Zabbix otrzyma informację o wyłączeniu hosta. W tym krótkim czasie host będzie nadal lokalnie zbierać dane dla aktywnych kontroli i próbować wysyłać je do serwera/proxy, jednak ponieważ host jest oznaczony jako *Wyłączony*, serwer/proxy odrzuci te dane. Po wyłączeniu hosta jego pozycje są natychmiast usuwane z pamięci podręcznej historii (z wyjątkiem ich ostatnich wartości, które są zachowywane dla dzienników).

Karta **IPMI** zawiera atrybuty zarządzania IPMI.

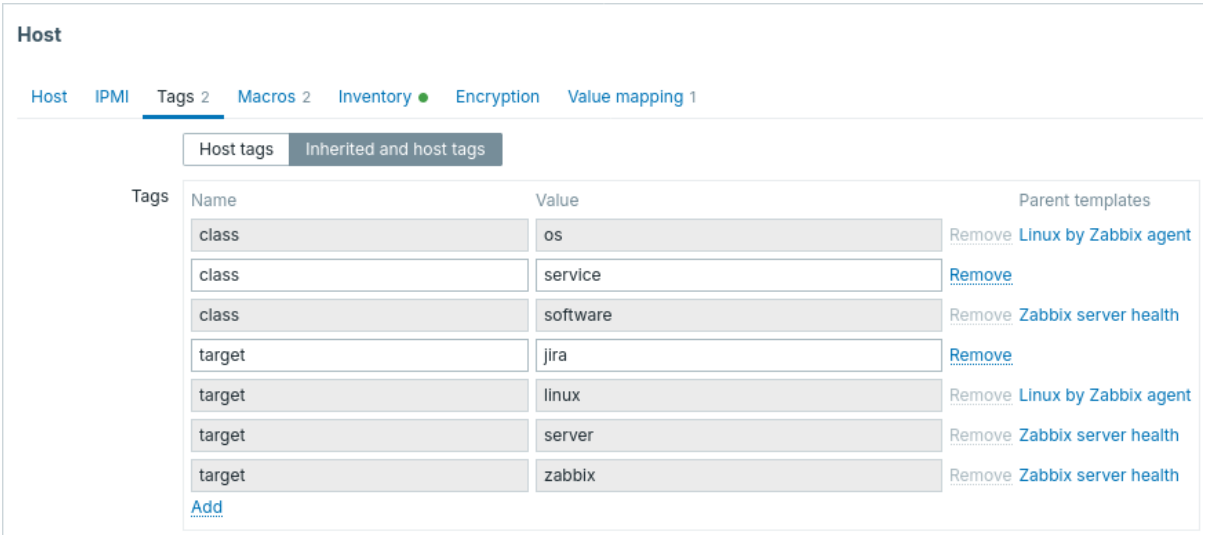
Parametr	Opis
Algorytm uwierzytelniania	Wybierz algorytm uwierzytelniania.
Poziom uprawnień	Wybierz poziom uprawnień.
Nazwa użytkownika	Nazwa użytkownika używana do uwierzytelniania. Można używać makr użytkownika.
Hasło	Hasło używane do uwierzytelniania. Można używać makr użytkownika.

Karta **Tagi** umożliwia zdefiniowanie tagów na poziomie hosta. Wszystkie problemy tego hosta zostaną otagowane wartościami wprowadzonymi tutaj.



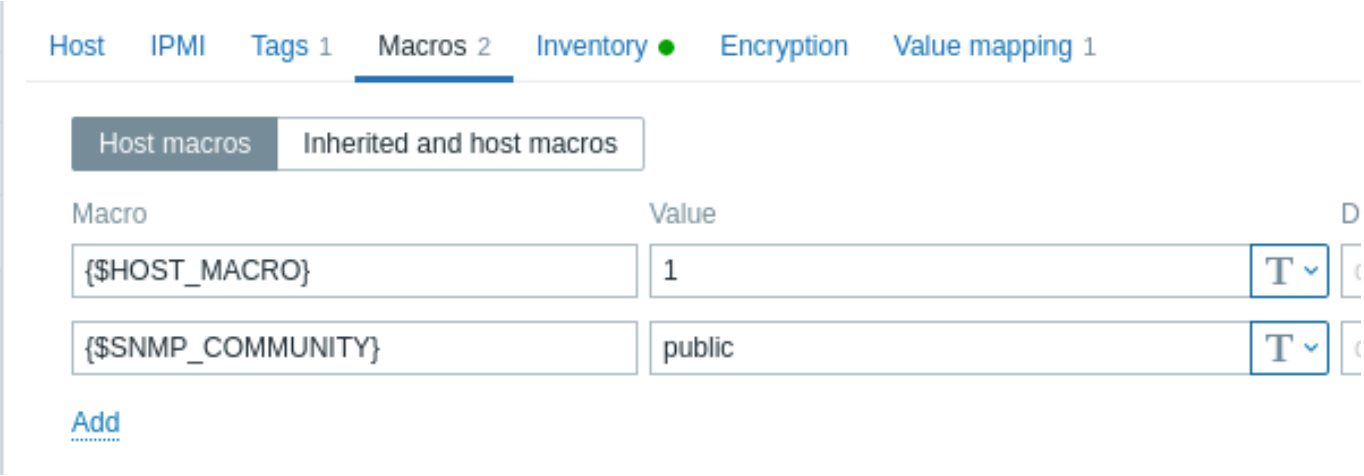
W tagach obsługiwane są makra użytkownika, makra `{INVENTORY.*}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.IP}`, `{HOST.PORT}` i `{HOST.ID}`.

Możesz również wyświetlić tutaj tagi na poziomie szablonu, wybierając opcję *Odziedziczone i tagi hosta*. W tym miejscu wyświetlane są wszystkie zdefiniowane tagi użytkownika hosta oraz ich źródło.

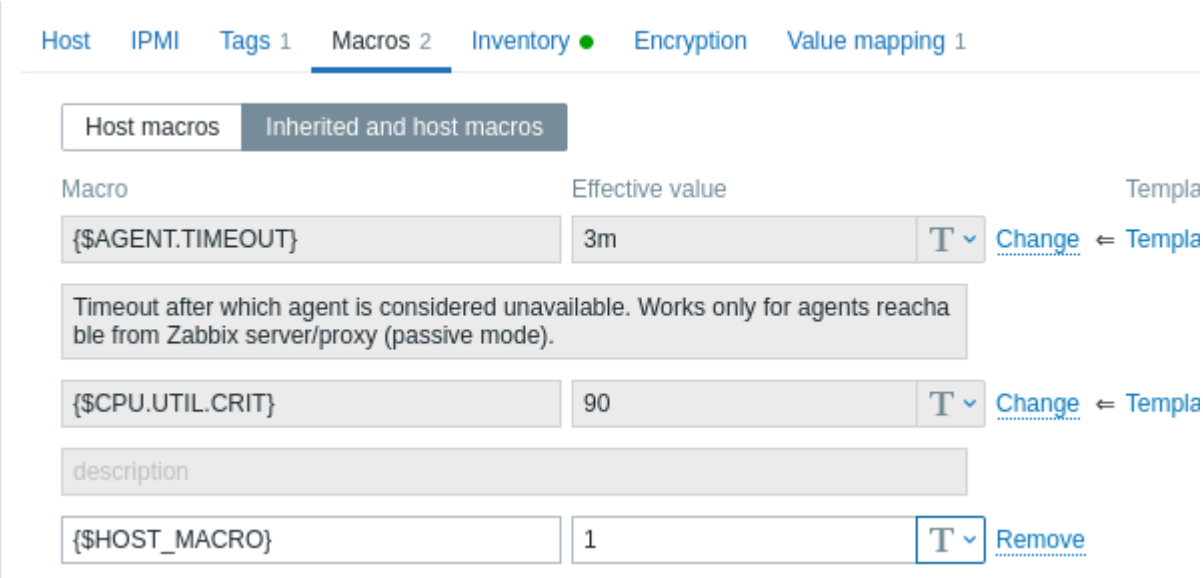


Dla wygody udostępniono odnośniki do odpowiednich szablonów.

Karta **Makra** umożliwia zdefiniowanie makr użytkownika na poziomie hosta w postaci par nazwa-wartość. Wartości makr mogą być przechowywane jako zwykły tekst, tekst poufny lub sekret Vault. Można również dodać opis.

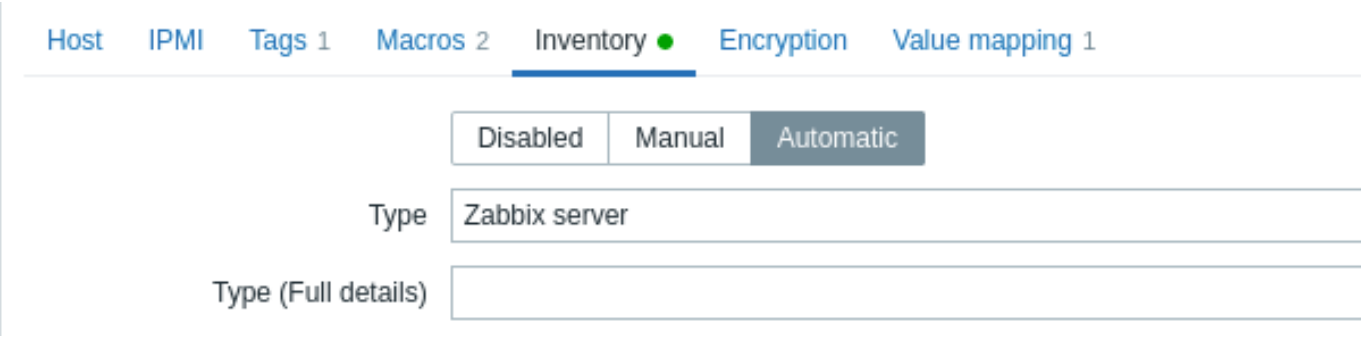


Możesz również wyświetlić tutaj makra użytkownika na poziomie szablonu i globalnym, wybierając opcję *Odziedziczone i makra hosta*. W tym miejscu wyświetlane są wszystkie zdefiniowane makra użytkownika hosta wraz z wartościami, do których się rozwijają, oraz ich źródłem.



Dla wygody udostępniono odnośniki do odpowiednich szablonów i globalnej konfiguracji makr. Możliwe jest również edytowanie makra szablonu/globalnego na poziomie hosta, co skutecznie tworzy kopię makra na hoście.

Karta **Inwentaryzacja** umożliwia ręczne wprowadzenie informacji o inwentaryzacji hosta. Możesz również włączyć *Automatyczne* uzupełnianie inwentaryzacji albo wyłączyć uzupełnianie inwentaryzacji dla tego hosta.



Jeśli inwentaryzacja jest włączona (ręcznie lub automatycznie), obok nazwy karty wyświetlana jest zielona kropka.

Szyfrowanie

Karta **Szyfrowanie** pozwala wymagać połączeń zaszyfrowanych z hostem.

Parameter	Description
Connections to host	W jaki sposób serwer Zabbix lub proxy łączy się z agentem Zabbix na hoście: bez szyfrowania (domyślnie), z użyciem PSK (pre-shared key) lub certyfikatu.

Parameter	Description
Connections from host	Wybierz, jaki typ połączeń jest dozwolony z hosta (tj. od agenta Zabbix i Zabbix sendera). Można wybrać kilka typów połączeń jednocześnie (przydatne podczas testowania i przełączania na inny typ połączenia). Domyślnie: "No encryption".
Issuer	Dozwolony wystawca certyfikatu. Certyfikat jest najpierw weryfikowany za pomocą CA (certificate authority). Jeśli jest poprawny i podpisany przez CA, pole *Issuer* może służyć do dalszego ograniczenia dozwolonego CA. To pole jest przeznaczone do użycia, jeśli Twoja instalacja Zabbix korzysta z certyfikatów od wielu CA. Jeśli to pole jest puste, akceptowana jest dowolna CA.
Subject	Dozwolony podmiot certyfikatu. Certyfikat jest najpierw weryfikowany za pomocą CA. Jeśli jest poprawny i podpisany przez CA, pole *Subject* może służyć do zezwolenia tylko na jedną wartość ciągu *Subject*. Jeśli to pole jest puste, akceptowany jest dowolny poprawny certyfikat podpisany przez skonfigurowaną CA.
PSK identity	Ciąg identyfikacyjny pre-shared key. Nie umieszczaj wrażliwych informacji w identyfikatorze PSK, ponieważ jest on przesyłany przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć.
PSK	Pre-shared key (ciąg szesnastkowy). Maksymalna długość: 512 cyfr szesnastkowych (PSK o rozmiarze 256 bajtów), jeśli Zabbix używa biblioteki GnuTLS lub OpenSSL, 64 cyfry szesnastkowe (PSK o rozmiarze 32 bajtów), jeśli Zabbix używa biblioteki mbed TLS (PolarSSL). Przykład: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Mapowanie wartości

Karta **Mapowanie wartości** umożliwia skonfigurowanie przyjaznej dla użytkownika reprezentacji danych pozycji w mapowaniach wartości.

Konfigurowanie grupy hostów

Przegląd

Grupy hostów są używane do logicznego grupowania hostów i przypisywania do nich uprawnień użytkowników.

Każdy host musi mieć przypisaną co najmniej jedną grupę hostów. Host może należeć do wielu grup hostów, a każda grupa hostów może zawierać wiele hostów.

Należy pamiętać, że w Zabbix wszystkie uprawnienia są oparte na grupach: grupach użytkowników, grupach hostów oraz grupach szablonów. Tak więc nawet jeśli pojedynczy użytkownik potrzebuje dostępu do pojedynczego hosta, jest on przyznawany przez dodanie użytkownika do grupy użytkowników i przypisanie tej grupie użytkowników uprawnień do grupy hostów zawierającej ten host.

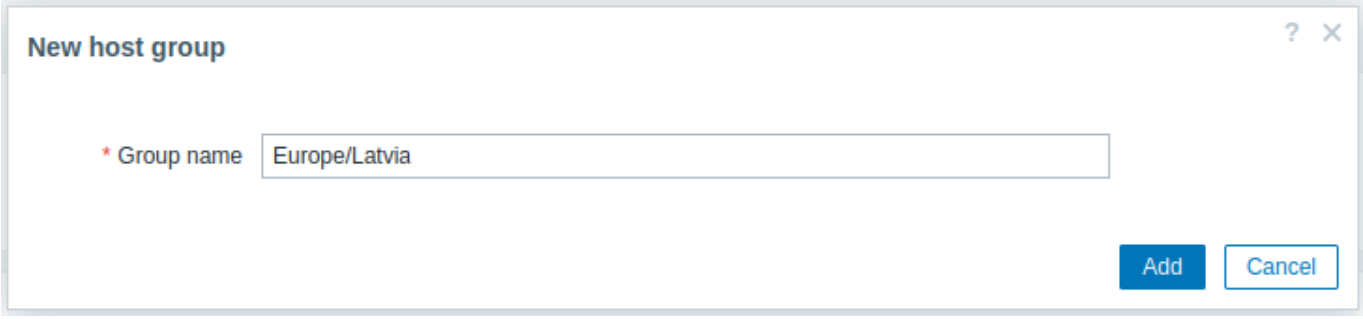
Konfiguracja

> **Attention:**
> Tylko użytkownicy z uprawnieniami Super admin mogą tworzyć grupy hostów.

W frontend Zabbix dostępne są dwie opcje tworzenia grupy hostów.

Opcja pierwsza:

1. Przejdź do *Zbieranie danych* > *Grupy hostów*.
2. Kliknij *Utwórz grupę hostów* w prawym górnym rogu ekranu.
3. Wprowadź nazwę grupy w formularzu.



Opcja druga: podczas konfigurowania hosta wpisz nazwę nieistniejącej grupy w polu *Grupy hostów*.

Po utworzeniu grupy hostów możesz kliknąć nazwę grupy na liście w *Zbieranie danych* > *Grupy hostów*, aby edytować nazwę grupy, sklonować grupę lub usunąć grupę.

Usunięcie grupy hostów usuwa tylko grupę logiczną, a nie hosty należące do tej grupy. Nie można usunąć grupy hostów, która jest jedyną grupą dla dowolnego istniejącego hosta.

Tworzenie podgrup hostów

Podgrupa hostów (lub zagnieżdżona grupa hostów) jest elementem podrzędnym nadrzędnej grupy hostów, która ją zawiera.

Podgrupę tworzy się przy użyciu ukośnika '/' w polu wprowadzania nazwy grupy, aby określić jej relację do grupy nadrzędnej lub grup nadrzędnych. Na przykład:

- wpisanie `Europe/Latvia` tworzy podgrupę `Europe/Latvia` w nadrzędnej grupie `Europe`.
- wpisanie `Europe/Latvia/Riga/Zabbix servers` tworzy odpowiednią podgrupę w zagnieżdżonych grupach nadrzędnych `Europe`, `Europe/Latvia`, `Europe/Latvia/Riga`.

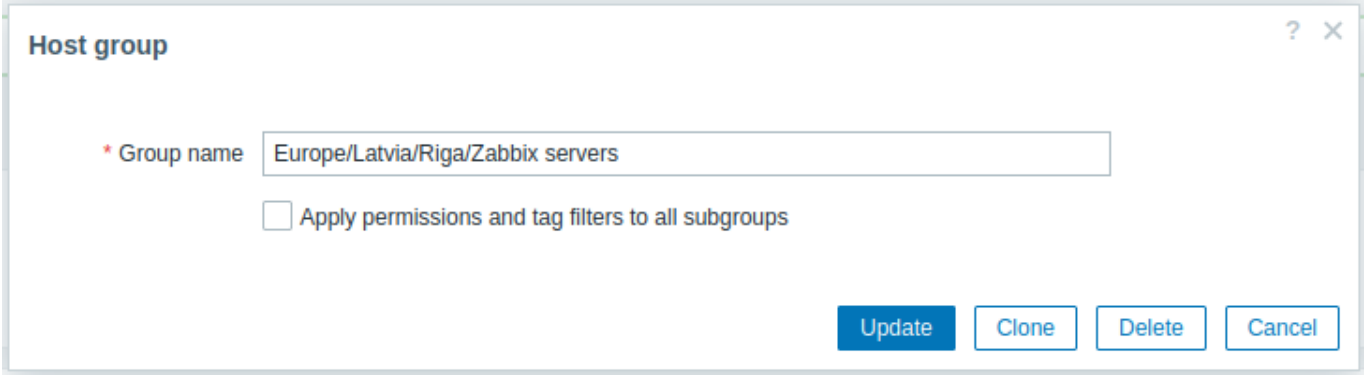
Podczas tworzenia podgrupy nie można używać ukośników na początku ani na końcu nazwy, ani kilku ukośników z rzędu. Escapowanie znaku '/' nie jest obsługiwane.

Nie jest wymagane tworzenie żadnych nadrzędnych grup hostów przed utworzeniem podgrupy. Możesz zdecydować, czy zacząć od utworzenia podgrupy (na przykład `Europe/Latvia`), czy dowolnej nadrzędnej grupy hostów (w naszym przykładzie `Europe`). Jeśli zaczniesz od utworzenia podgrupy, nadrzędne grupy hostów **nie** zostaną utworzone automatycznie.

Uprawnienia do grup hostów

- Podczas tworzenia podgrupy w istniejącej nadrzędnej grupie hostów (na przykład tworzenia `Europe/Latvia`, gdy `Europe` już istnieje), uprawnienia grupy użytkowników do podgrupy są dziedziczone z grupy nadrzędnej.
- Podczas tworzenia nadrzędnej grupy hostów dla istniejącej podgrupy (na przykład tworzenia `Europe`, gdy `Europe/Latvia` już istnieje), nie są ustawiane żadne uprawnienia dla grupy nadrzędnej.

Podczas edycji dowolnej grupy hostów można również ustawić dodatkową opcję *Zastosuj uprawnienia i filtry tagów do wszystkich podgrup*.



Zaznaczenie tego pola wyboru i kliknięcie *Update* spowoduje zastosowanie tego samego poziomu uprawnień i filtrów tagów do wszystkich bieżących i przyszłych podgrup edytowanej grupy hostów.

Jeśli więc jakimkolwiek grupom użytkowników przyznano różne uprawnienia do podgrup edytowanej grupy hostów, zaznaczenie pola wyboru nada wszystkim bieżącym i przyszłym podgrupom takie same uprawnienia użytkowników oraz uprawnienia oparte na tagach jak w grupie będącej edytowana.

Należy pamiętać, że ta opcja nie jest zapisywana w bazie danych i zastąpi istniejące uprawnienia. Wszelkie zmiany wprowadzone za pomocą tej opcji można cofnąć tylko ręcznie.

Inwentarz

Przegląd

Możesz przechowywać inwentarz urządzeń sieciowych w Zabbix.

W Zabbix frontend dostępne jest specjalne menu *Inwentarz*. Jednak początkowo nie zobaczysz tam żadnych danych i nie jest to miejsce do ich wprowadzania. Tworzenie danych inwentarza odbywa się ręcznie podczas konfigurowania hosta lub automatycznie przy użyciu niektórych opcji automatycznego uzupełniania.

Tworzenie inwentarza

Tryb ręczny

Podczas konfigurowania hosta na karcie *Inwentaryzacja* można wprowadzić takie szczegóły, jak typ urządzenia, numer seryjny, lokalizacja, osoba odpowiedzialna, adresy URL itp. — dane, które zostaną użyte do uzupełnienia informacji inwentaryzacyjnych.

Jeśli w informacjach inwentaryzacyjnych hosta znajduje się adres URL rozpoczynający się od „http" lub „https", w sekcji *Inwentaryzacja* zostanie on wyświetlony jako klikalny link.

Tryb automatyczny

Inwentaryzację hosta można również wypełniać automatycznie. Aby to działało, podczas konfigurowania hosta tryb inwentaryzacji na karcie *Inwentaryzacja* musi być ustawiony na *Automatyczny*.

Następnie można skonfigurować pozycje hosta, aby wypełniały dowolne pole inwentaryzacji hosta swoją wartością, wskazując docelowe pole za pomocą odpowiedniego atrybutu (o nazwie *Pozycja wypełni pole inwentaryzacji hosta*) w konfiguracji pozycji.

Pozycje szczególnie przydatne do automatycznego zbierania danych inwentaryzacyjnych:

- system.hw.chassis[full|type|vendor|model|serial] - domyślnie jest to [full], wymagane uprawnienia root
- system.hw.cpu[all|cpunum,full|maxfreq|vendor|model|curfreq] - domyślnie jest to [all,full]
- system.hw.devices[pci|usb] - domyślnie jest to [pci]
- system.hw.macaddr[interface,short|full] - domyślnie jest to [all,full], interface jest wyrażeniem regularnym
- system.sw.arch
- system.sw.os[name|short|full] - domyślnie jest to [name]
- system.sw.packages[regexp,manager,short|full] - domyślnie jest to [all,all,full].

Wybór trybu inwentaryzacji

Tryb inwentaryzacji można wybrać w formularzu konfiguracji hosta.

Domyślny tryb inwentaryzacji dla nowych hostów jest wybierany na podstawie ustawienia *Domyślny tryb inwentaryzacji hosta* w sekcji *Administracja* > *Ogólne* > *Inne*.

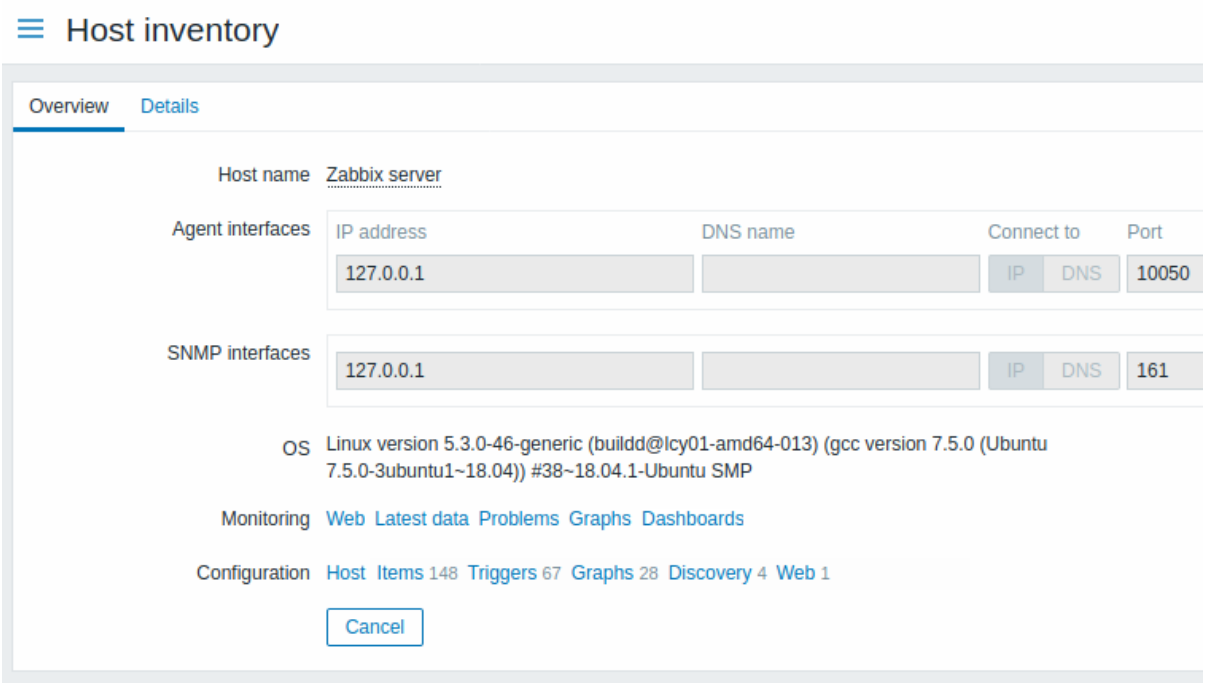
W przypadku hostów dodanych w wyniku wykrywania sieci lub działań automatycznej rejestracji można zdefiniować operację *Ustaw tryb inwentaryzacji hosta*, wybierając tryb ręczny lub automatyczny. Ta operacja zastępuje ustawienie *Domyślny tryb inwentaryzacji hosta*.

Przegląd inwentarza

Szczegóły wszystkich istniejących danych inwentarza są dostępne w menu *Inventory*.

W *Inventory* > *Overview* można uzyskać liczbę hostów według różnych pól inwentarza.

W *Inventory* > *Hosts* można zobaczyć wszystkie hosty, które mają informacje inwentarza. Kliknięcie nazwy hosta spowoduje wyświetlenie szczegółów inwentarza w formularzu.



Karta **Overview** pokazuje:

Parameter	Description
Host name	Nazwa hosta. Kliknięcie nazwy otwiera menu ze skryptami zdefiniowanymi dla hosta. Nazwa hosta jest wyświetlana z pomarańczową ikoną, jeśli host znajduje się w trybie konserwacji.
Visible name	Widoczna nazwa hosta (jeśli zdefiniowana).
Host (Agent, SNMP, JMX, IPMI) interfaces	Ten blok zawiera szczegóły interfejsów skonfigurowanych dla hosta.
OS	Pole inwentarza systemu operacyjnego hosta (jeśli zdefiniowane).
Hardware	Pole inwentarza sprzętu hosta (jeśli zdefiniowane).
Software	Pole inwentarza oprogramowania hosta (jeśli zdefiniowane).
Description	Opis hosta.
Monitoring	Odnośniki do sekcji monitorowania z danymi dla tego hosta: *Web*, *Latest data*, *Problems*, *Graphs*, *Dashboards*.
Configuration	Odnośniki do sekcji konfiguracji dla tego hosta: *Host*, *Items*, *Triggers*, *Graphs*, *Discovery*, *Web*. Liczba skonfigurowanych obiektów jest podana po każdym odnośniku.

Karta **Details** pokazuje wszystkie wypełnione pola inwentarza (nie są puste).

Makra inwentarza

Dostępne są makra inwentarza hosta `{INVENTORY.*}`, które można używać w powiadomieniach, na przykład:

”Serwer w {INVENTORY.LOCATION1} ma problem, osobą odpowiedzialną jest {INVENTORY.CONTACT1}, numer telefonu {INVENTORY.POC.PRIMARY.PHONE.A1}.”

Więcej informacji można znaleźć na stronie Obsługiwane makra.

Masowa aktualizacja

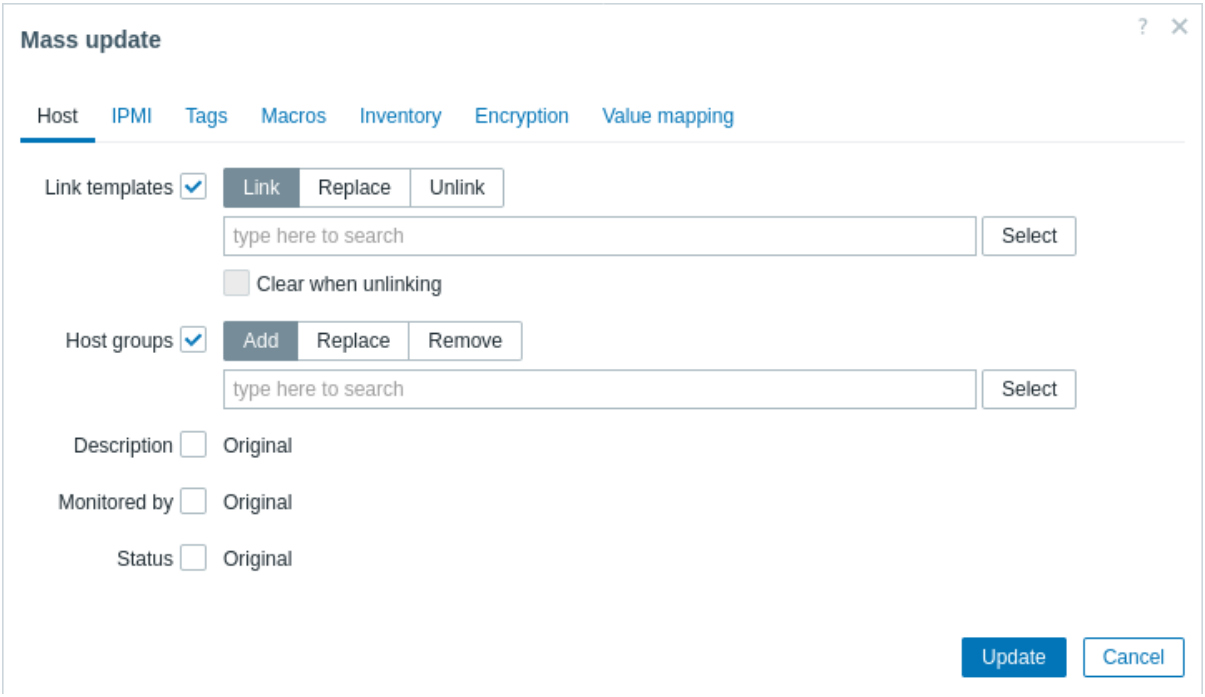
Przegląd

Czasami możesz chcieć naraz zmienić pewien atrybut dla wielu hostów. Zamiast otwierać każdy host do edycji z osobna, możesz skorzystać z funkcji aktualizacji masowej.

Aktualizacja zbiorcza

Aby zbiorczo zaktualizować niektóre hosty, wykonaj następujące czynności:

- Zaznacz pola wyboru przed hostami, które chcesz zaktualizować, na liście hostów
- Kliknij opcję *Aktualizacja zbiorcza* poniżej listy
- Przejdź do karty zawierającej wymagane atrybuty (*Host*, *IPMI*, *Znaczniki*, *Makra*, *Inwentaryzacja*, *Szyfrowanie* lub *Mapowanie wartości*)
- Zaznacz pola wyboru dowolnych atrybutów, które chcesz zaktualizować, i wprowadź dla nich nowe wartości



Po wybraniu odpowiedniego przycisku aktualizacji powiązań z **szablonami** dostępne są następujące opcje:

- *Połącz* - określ dodatkowe szablony, które mają zostać połączone
- *Zastąp* - określ szablony, które mają zostać połączone, jednocześnie odłączając wszystkie szablony wcześniej połączone z hostami
- *Odłącz* - określ szablony, które mają zostać odłączone

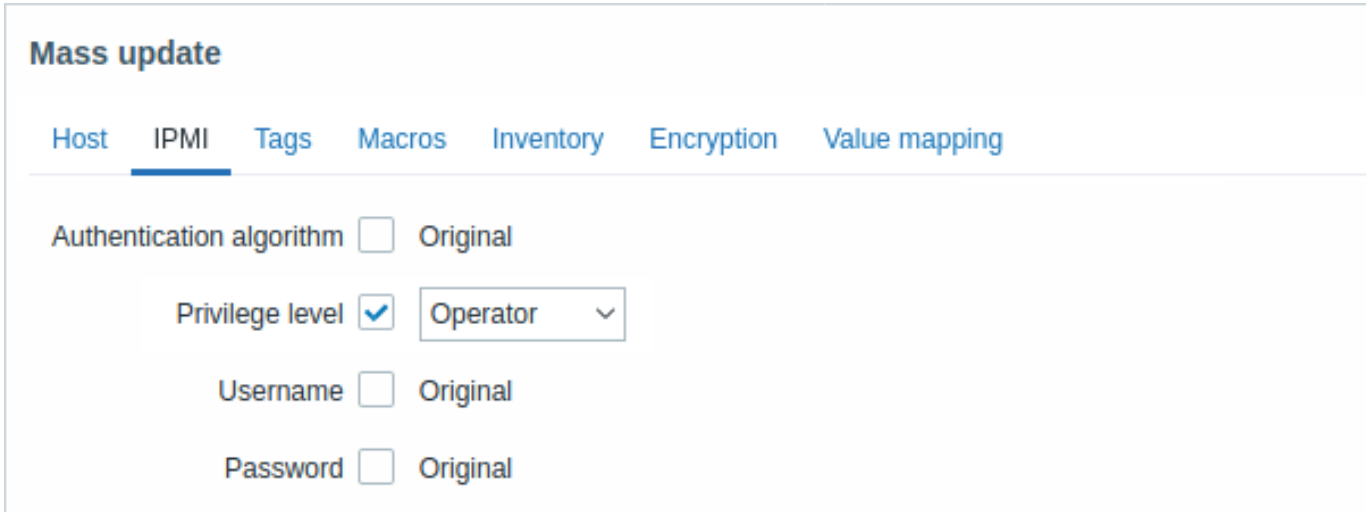
Aby określić szablony do połączenia lub odłączenia, zacznij wpisywać nazwę szablonu w polu autouzupełniania, aż pojawi się lista rozwijana z pasującymi szablonami. Przewiń listę, aby wybrać wymagany szablon.

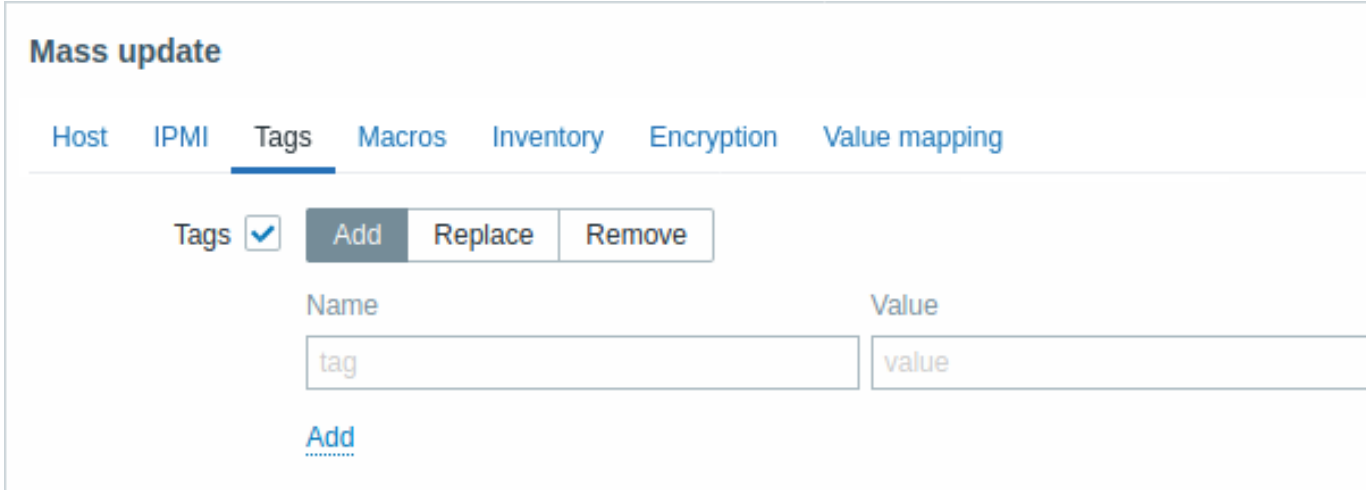
Opcja *Wyczyść podczas odłączania* umożliwia nie tylko odłączenie wcześniej połączonych szablonów, ale także usunięcie wszystkich odziedziczonych z nich encji (pozycji, wyzwalaczy, wykresów itd.).

Po wybraniu odpowiedniego przycisku aktualizacji **grup hostów** dostępne są następujące opcje:

- *Dodaj* - określ dodatkowe grupy hostów spośród istniejących lub wprowadź całkowicie nowe grupy hostów dla hostów;
- *Zastąp* - usuń hosta ze wszystkich istniejących grup hostów i zastąp je grupą lub grupami określonymi w tym polu (istniejącymi lub nowymi grupami hostów);
- *Usuń* - usuń określone grupy hostów z hostów.

Pola te obsługują autouzupełnianie - rozpoczęcie wpisywania powoduje wyświetlenie listy rozwijanej z pasującymi grupami hostów. Jeśli grupa hostów jest nowa, również pojawi się na liście rozwijanej i będzie oznaczona ciągiem *(new)*. Przewiń listę, aby ją wybrać.

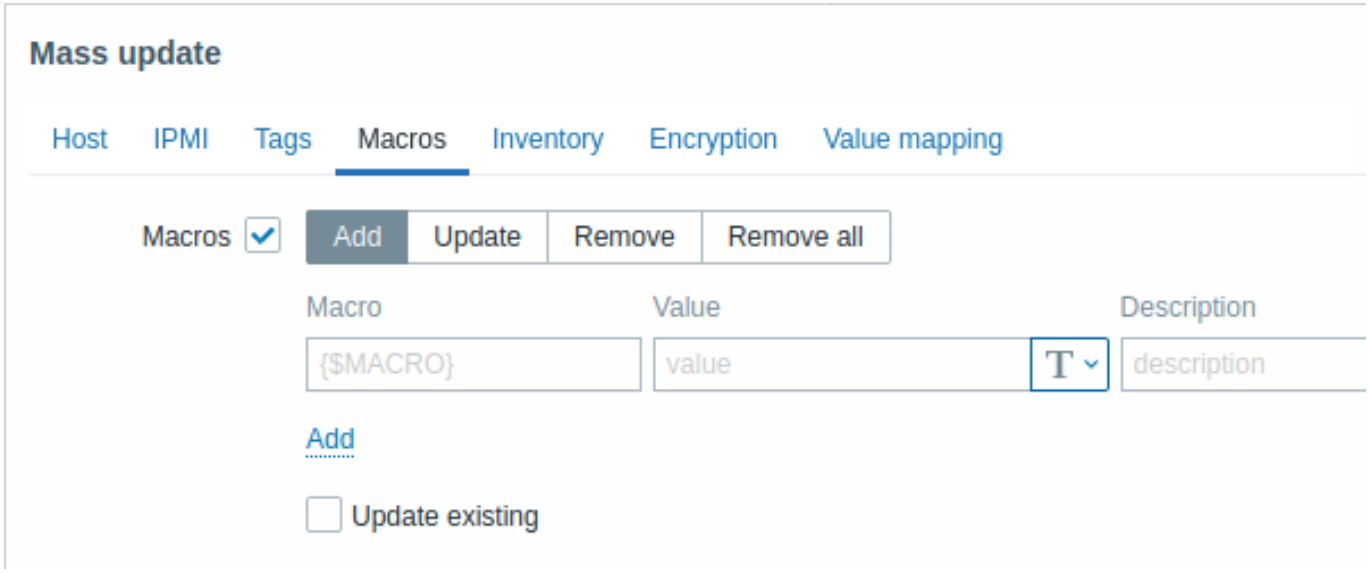




Po wybraniu odpowiedniego przycisku aktualizacji znaczników dostępne są następujące opcje:

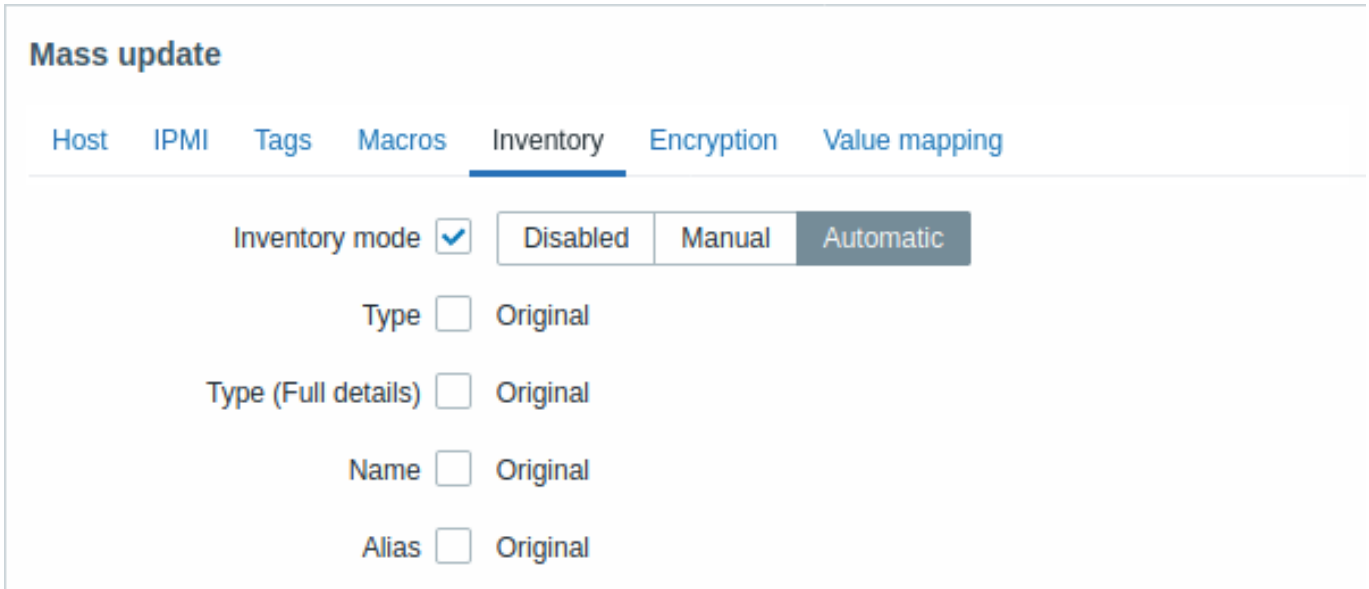
- *Dodaj* - dodaj nowe znaczniki do hostów;
- *Zastąp* - usuń wszystkie istniejące znaczniki z hosta i zastąp je znacznikiem lub znacznikami określonymi poniżej;
- *Usuń* - usuń określone znaczniki z hostów.

Makra użytkownika, makra `{INVENTORY.*}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.IP}`, `{HOST.PORT}` i `{HOST.ID}` są obsługiwane w znacznikach. Należy pamiętać, że znaczniki o tej samej nazwie, ale różnych wartościach, nie są uznawane za „duplikaty” i można je dodać do tego samego hosta.

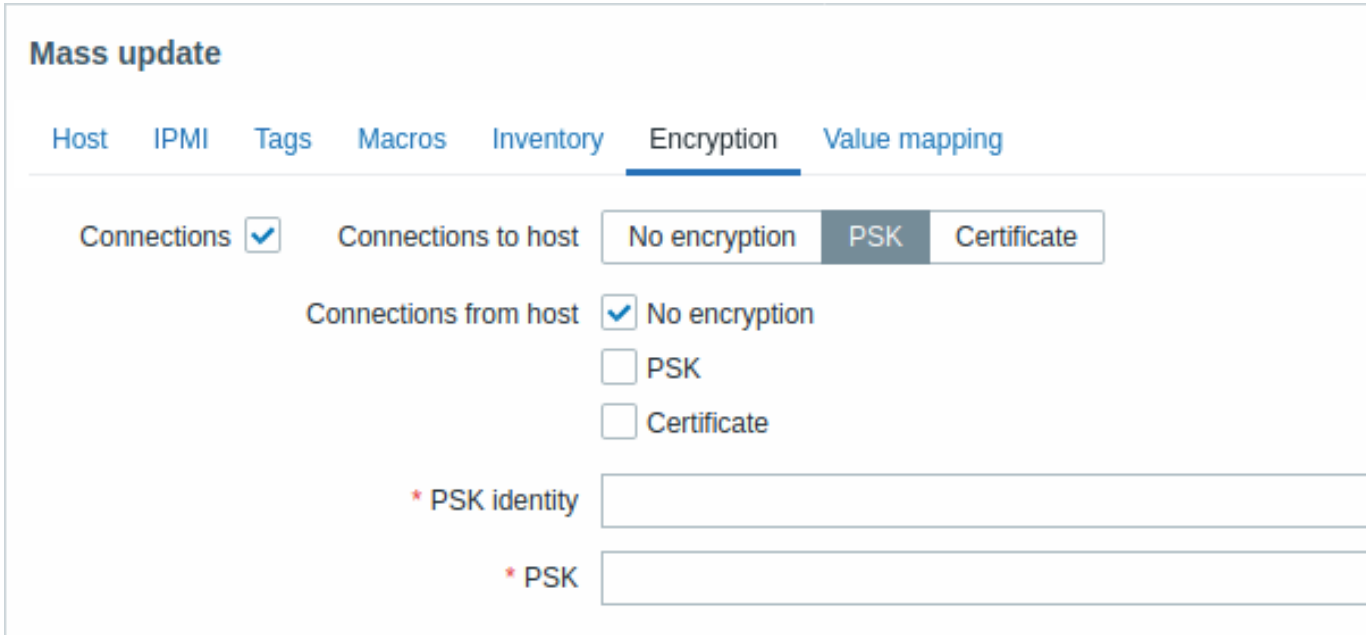


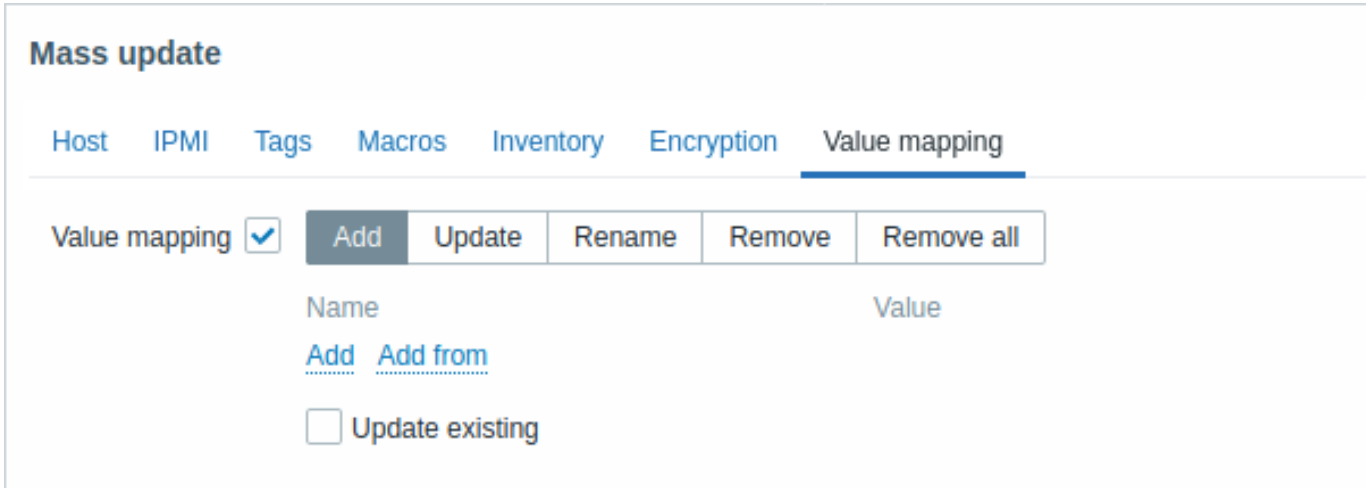
Po wybraniu odpowiedniego przycisku aktualizacji makr dostępne są następujące opcje:

- *Dodaj* - określ dodatkowe makra użytkownika dla hostów. Jeśli pole wyboru *Aktualizuj istniejące* jest zaznaczone, wartość, typ i opis określonej nazwy makra zostaną zaktualizowane. Jeśli pole nie jest zaznaczone, a makro o tej nazwie już istnieje na hoście lub hostach, nie zostanie zaktualizowane.
- *Aktualizuj* - zastąp wartości, typy i opisy makr określonych na tej liście. Jeśli pole wyboru *Dodaj brakujące* jest zaznaczone, makro, które wcześniej nie istniało na hoście, zostanie dodane jako nowe makro. Jeśli pole nie jest zaznaczone, zostaną zaktualizowane tylko makra, które już istnieją na hoście.
- *Usuń* - usuń określone makra z hostów. Jeśli pole wyboru *Z wyjątkiem zaznaczonych* jest zaznaczone, wszystkie makra z wyjątkiem określonych na liście zostaną usunięte. Jeśli pole nie jest zaznaczone, zostaną usunięte tylko makra określone na liście.
- *Usuń wszystkie* - usuń wszystkie makra użytkownika z hostów. Jeśli pole wyboru *Potwierdzam usunięcie wszystkich makr* nie jest zaznaczone, otworzy się nowe okno podręczne z prośbą o potwierdzenie usunięcia wszystkich makr.



Aby można było zbiorczo aktualizować pola inwentaryzacji, *Tryb inwentaryzacji* powinien być ustawiony na „Ręczny” lub „Automatyczny”.





W przypadku aktualizacji map wartości dostępne są przyciski z następującymi opcjami:

- *Dodaj* - dodaj mapy wartości do hostów. Jeśli zaznaczysz opcję *Aktualizuj istniejące*, wszystkie właściwości mapy wartości o tej nazwie zostaną zaktualizowane. W przeciwnym razie, jeśli mapa wartości o tej nazwie już istnieje, nie zostanie zaktualizowana.
- *Aktualizuj* - zaktualizuj istniejące mapy wartości. Jeśli zaznaczysz opcję *Dodaj brakujące*, mapa wartości, która wcześniej nie istniała na hoście, zostanie dodana jako nowa mapa wartości. W przeciwnym razie zostaną zaktualizowane tylko mapy wartości, które już istnieją na hoście.
- *Zmień nazwę* - nadaj nową nazwę istniejącej mapie wartości
- *Usuń* - usuń określone mapy wartości z hostów. Jeśli zaznaczysz opcję *Z wyjątkiem zaznaczonych*, wszystkie mapy wartości zostaną usunięte **z wyjątkiem** tych, które zostały określone.
- *Usuń wszystkie* - usuń wszystkie mapy wartości z hostów. Jeśli pole wyboru *Potwierdzam usunięcie wszystkich map wartości* nie jest zaznaczone, otworzy się nowe okno podręczne z prośbą o potwierdzenie usunięcia.

Po wprowadzeniu wszystkich wymaganych zmian kliknij opcję *Aktualizuj*. Atrybuty zostaną odpowiednio zaktualizowane dla wszystkich wybranych hostów.

Pozycje

#####Przegląd

Pozycja to indywidualna metryka.

Pozycje są używane do zbierania danych. Po skonfigurowaniu hosta, musisz dodać pozycje, aby uzyskać rzeczywiste dane. Jednym ze sposobów szybkiego dodawania wielu pozycji jest dołączenie jednego z predefiniowanych szablonów do hosta. Jednakże w celu zoptymalizowania wydajności systemu może zaistnieć potrzeba dostosowania szablonów tak, aby zawierały liczbę pozycji i częstotliwość monitorowania, o ile to konieczne.

Aby określić, jakie dane zbierać z hosta, użyj klucza pozycji. Na przykład, pozycja o kluczu **system.cpu.load** zbierze dane o obciążeniu procesora, podczas gdy pozycja o kluczu **net.if.in** zbierze informacje o przychodzącym ruchu sieciowym.

Dodatkowe parametry można określić w nawiasach kwadratowych po nazwie klucza. Na przykład, system.cpu.load**[avg5]** zwróci średnie obciążenie procesora (load) z ostatnich 5 minut, podczas gdy net.if.in**[eth0]** pokaże ilość przychodzącego ruchu na interfejsie "eth0".

Note:
Zobacz poszczególne sekcje typów pozycji dla wszystkich obsługiwanych typów pozycji i kluczy pozycji.

Przejdź do tworzenia i konfigurowania pozycji.

Tworzenie pozycji

Przegląd

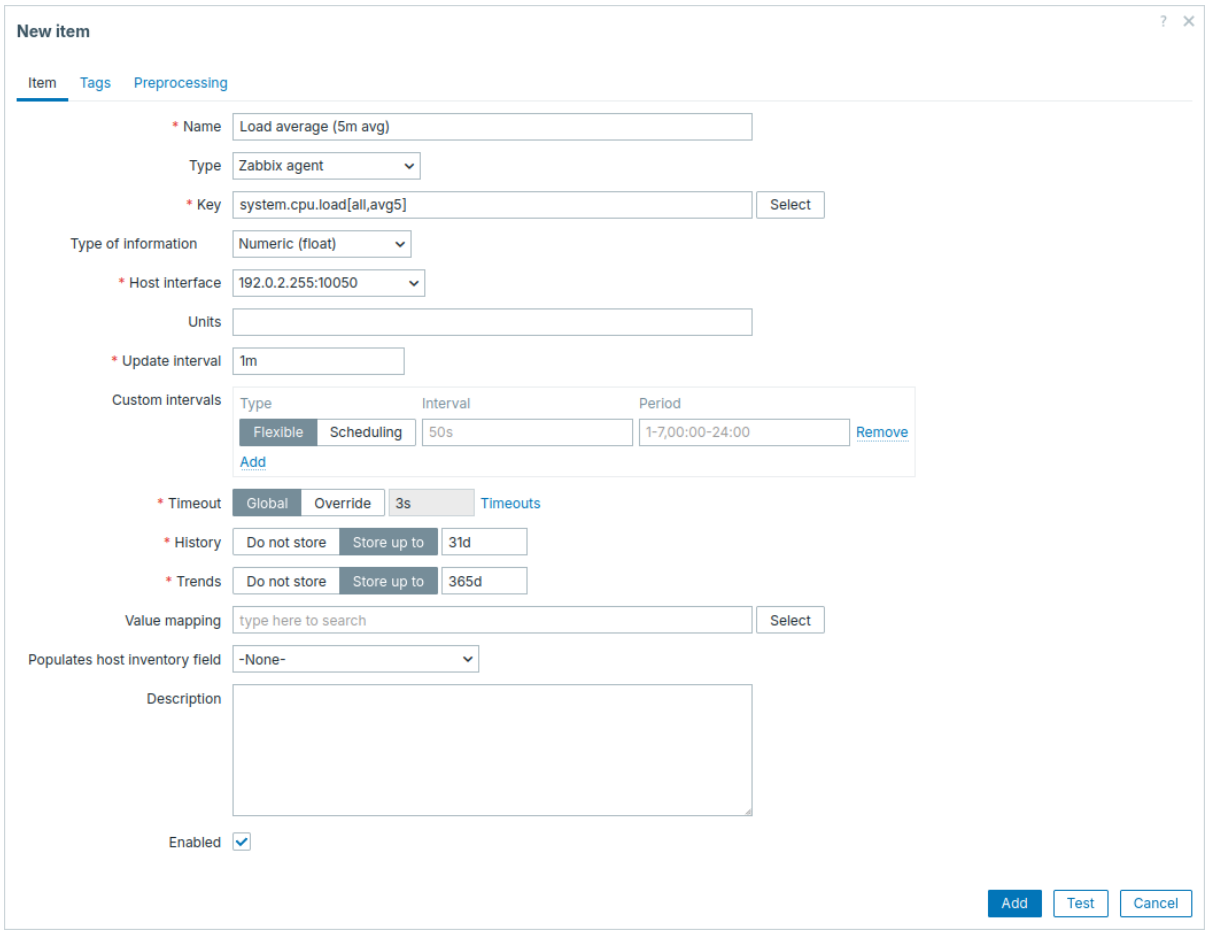
Aby utworzyć pozycję w frontend Zabbix, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Hosts*.
2. Kliknij *Items* w wierszu host.
3. Kliknij *Create item* w prawym górnym rogu ekranu.
4. Wprowadź parametry pozycji w formularzu.

Alternatywnie możesz utworzyć pozycję, otwierając istniejącą pozycję, klikając przycisk *Clone* i zapisując ją pod inną nazwą.

Konfiguracja

Karta **Pozycja** zawiera ogólne atrybuty pozycji.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa	Nazwa pozycji. Obsługiwane są makra użytkownika.
Typ	Typ pozycji. Zobacz poszczególne sekcje dotyczące typów pozycji.

Parametr	Opis
Klucz	Klucz pozycji (do 2048 znaków). Obsługiwane klucze pozycji można znaleźć w poszczególnych sekcjach dotyczących typów pozycji. Klucz musi być unikalny w obrębie pojedynczego hosta. Jeśli typ klucza to „Zabbix agent”, „Zabbix agent (active)” lub „Simple check”, wartość klucza musi być obsługiwana przez Zabbix agent lub Zabbix server. Zobacz także: prawidłowy format klucza.
Typ informacji	Typ danych używany do sprawdzania wartości pozycji i przechowywania jej w bazie danych, również po wykonaniu dowolnych konwersji: **Liczbowe (bez znaku)** - 64-bitowa liczba całkowita bez znaku; **Liczbowe (zmiennoprzecinkowe)** - 64-bitowa liczba zmiennoprzecinkowa; **Znakowe** - krótki tekst; **Dziennik** - długi tekst z właściwościami dziennika lub bez nich (znacznik czasu, źródło, ważność, identyfikator zdarzenia dziennika); **Tekstowe** - długi tekst; **Binarne** - liczba binarna (obsługiwana tylko dla pozycji zależnych); **JSON** - strukturyzowane dane JSON, przechowywane natywnie w bazie danych (nieobsługiwane dla pozycji obliczanych). Pozycje zwracające wartości typu binarnego lub JSON nie są obsługiwane w formułach pozycji obliczanych ani wyrażeniach wyzwalaczy. Limity przechowywania dla poszczególnych typów danych i backendów baz danych opisano w sekcji Limity danych pozycji. W przypadku pozycji zwracających dane w jednym określonym formacie odpowiedni typ danych jest wybierany automatycznie — na przykład `system.cpu.load` i `system.cpu.util` zwracają 64-bitową liczbę zmiennoprzecinkową, dlatego automatycznie wybierany jest typ *Liczbowe (zmiennoprzecinkowe)*.
Interfejs hosta	Wybierz interfejs hosta. To pole jest dostępne podczas edycji pozycji na poziomie hosta.
Jednostki	Jeśli ustawiono symbol jednostki, Zabbix stosuje przetwarzanie końcowe do odebranej wartości pozycji i wyświetla ją z określonym przyrostkiem jednostki. Obsługiwane symbole jednostek ze specjalnym formatowaniem (oraz przykłady: odebrana wartość → wyświetlana wartość): **B** - bajty (1024 → 1 KB) **Bps** - bajty na sekundę (1024 → 1 KBps) **s** - sekundy, wyświetlane przy użyciu maksymalnie trzech największych niezerowych jednostek czasu (881764 → 10d 4h 56m) **uptime** - czas działania w formacie hh:mm:ss lub N dni, hh:mm:ss (881764 → 10 days, 04:56:04) **unixtime** - znacznik czasu Unix, formatowany jako yyyy.mm.dd hh:mm:ss (881764 → 1970-01-11 04:56:04 AM); aby formatowanie było prawidłowe, odebrana wartość pozycji musi mieć typ *Liczbowe (bez znaku)*. W przypadku innych jednostek (takich jak Hz, W itd.), jeśli odebrana wartość przekracza 1000, jest dzielona przez 1000 i wyświetlana z odpowiednim przedrostkiem (5000 → 5 KHz, 881764 → 881.76 KW). Jeśli przed symbolem jednostki znajduje się `!`, konwersja jednostki i dodawanie przedrostka są wyłączone (1024 !B → 1024 B, 61 !s → 61 s). Więcej przykładów i informacji o symbolach jednostek oraz konwersji jednostek zawiera sekcja Przyrostki wartości pozycji.
Interwał aktualizacji	Pobieraj nową wartość tej pozycji co N sekund. Maksymalny dozwolony interwał aktualizacji wynosi 86400 sekund (1 dzień). Obsługiwane są przyrostki czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. Pojedyncze makro musi wypełniać całe pole. Wiele makr w jednym polu ani makra połączone z tekstem nie są obsługiwane. *Uwaga*: Interwał aktualizacji można ustawić na „0” tylko wtedy, gdy istnieją niestandardowe interwały z wartością różną od zera. Jeśli ustawiono „0” i istnieje niestandardowy interwał (elastyczny lub zaplanowany) z wartością różną od zera, odpytywanie pozycji będzie odbywać się przez czas trwania niestandardowego interwału. *Uwaga*: pierwsze odpytywanie pozycji po jej aktywowaniu lub po zmianie interwału aktualizacji może nastąpić wcześniej niż wynika to ze skonfigurowanej wartości. Nowe pozycje zostaną sprawdzone w ciągu 60 sekund od ich utworzenia, chyba że mają zaplanowany lub elastyczny interwał aktualizacji, a *Interwał aktualizacji* jest ustawiony na 0. Istniejącą pasywną pozycję można natychmiast odpytać o wartość, klikając przycisk *Wykonaj teraz* button.

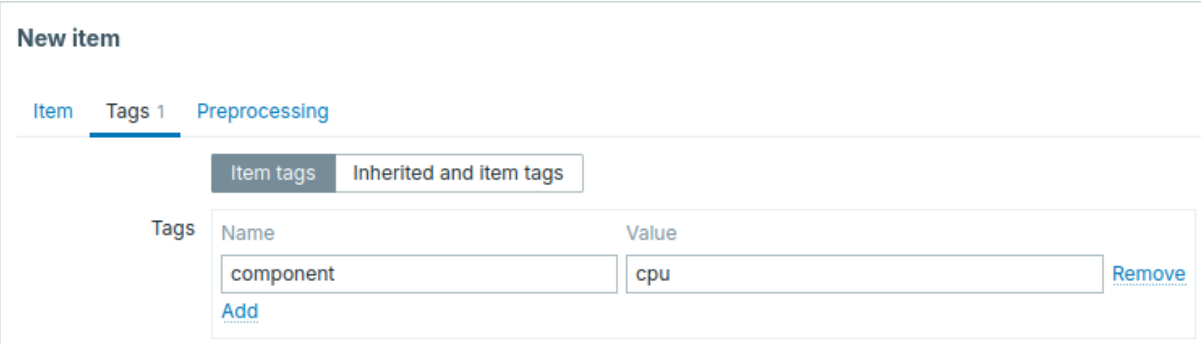
Parametr	Opis
Niestandardowe interwały	Możesz utworzyć niestandardowe reguły sprawdzania pozycji: **Elastyczny** - utworzenie wyjątku dla *Interwału aktualizacji* (interwał z inną częstotliwością). **Planowanie** - utworzenie niestandardowego harmonogramu odpytywania. Szczegółowe informacje zawiera sekcja Niestandardowe interwały. W polu *Interwał* obsługiwane są przyrostki czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. Pojedyncze makro musi wypełniać całe pole. Wiele makr w jednym polu ani makra połączone z tekstem nie są obsługiwane.
Limit czasu	Ustaw limit czasu sprawdzania pozycji (dostępny dla obsługiwanych typów pozycji). Wybierz opcję limitu czasu: **Globalny** - używany jest globalny limit czasu serwera proxy (wyświetlany w wyszarzonym polu *Limit czasu*). **Zastąpienie** - używany jest niestandardowy limit czasu (ustawiany w polu *Limit czasu*; dozwolony zakres: 1 - 600s). Obsługiwane są przyrostki czasu, np. 30s, 1m, oraz makra użytkownika. Kliknięcie łącza *Limity czasu* umożliwia skonfigurowanie limitów czasu proxy lub globalnych (jeśli proxy nie jest używany). Łącze *Limity czasu* jest widoczne tylko dla użytkowników typu *Super admin* z uprawnieniami do sekcji frontend *Administracja* > *Ogólne* lub *Administracja* > *Proxy*.
Historia	Wybierz jedną z opcji: **Nie przechowuj** - historia pozycji nie jest przechowywana. Przydatne w przypadku pozycji głównych, jeśli tylko pozycje zależne muszą zachowywać historię. To ustawienie nie może zostać zastąpione przez globalne ustawienia housekeepera. **Przechowuj przez** - określ czas przechowywania szczegółowej historii w bazie danych (od 1 godziny do 25 lat). Starsze dane zostaną usunięte przez housekeepera. Wartość jest przechowywana w sekundach. Obsługiwane są przyrostki czasu, np. 2h, 1d. Obsługiwane są makra użytkownika. Wartość *Przechowuj przez* można globalnie zastąpić w sekcji *Administracja* > *Housekeeping*. Jeśli istnieje globalne ustawienie zastępujące, wyświetlana jest pomarańczowa ikona informacji  . Po umieszczeniu na niej wskaźnika myszy wyświetlany jest komunikat ostrzegawczy, np. *Zastąpione przez globalne ustawienia housekeepera (1d)*. Zaleca się przechowywanie zarejestrowanych wartości przez możliwie najkrótszy czas, aby zmniejszyć rozmiar historii wartości w bazie danych. Zamiast przechowywać długą historię wartości, można przechowywać dane trendów przez dłuższy czas. Zobacz także Historia i trendy.
Trendy	Wybierz jedną z opcji: **Nie przechowuj** - trendy nie są przechowywane. To ustawienie nie może zostać zastąpione przez globalne ustawienia housekeepera. **Przechowuj przez** - określ czas przechowywania zagregowanej historii (godzinowe wartości min, maks., średnia, liczba) w bazie danych (od 1 dnia do 25 lat). Starsze dane zostaną usunięte przez housekeepera. Wartość jest przechowywana w sekundach. Obsługiwane są przyrostki czasu, np. 24h, 1d. Obsługiwane są makra użytkownika. Wartość *Przechowuj przez* można globalnie zastąpić w sekcji *Administracja* > *Housekeeping*. Jeśli istnieje globalne ustawienie zastępujące, wyświetlana jest pomarańczowa ikona informacji  . Po umieszczeniu na niej wskaźnika myszy wyświetlany jest komunikat ostrzegawczy, np. *Zastąpione przez globalne ustawienia housekeepera (7d)*. *Uwaga:* Przechowywanie trendów nie jest dostępne dla danych nieliczbowych - znakowych, dziennika i tekstowych. Zobacz także Historia i trendy.
Mapowanie wartości	Zastosuj mapowanie wartości do tej pozycji. Mapowanie wartości nie zmienia odebranych wartości, służy wyłącznie do wyświetlania danych. Działa z pozycjami typu *Liczbowe (bez znaku)*, *Liczbowe (zmiennoprzecinkowe)* i *Znakowe*. Na przykład „Stany usług Windows”.

Parametr	Opis
Format czasu dziennika	Dostępne tylko dla pozycji typu **Dziennik**. Obsługiwane symbole zastępcze: **y**: *Rok (1970-2038)*. **M**: *Miesiąc (01-12)*. **d**: *Dzień (01-31)*. **h**: *Godzina (00-23)*. **m**: *Minuta (00-59)*. **s**: *Sekunda (00-59)*. Jeśli pole pozostanie puste, znacznik czasu zostanie ustawiony na 0 w czasie Unix, co oznacza 1 stycznia 1970 r. Rozważmy na przykład następujący wiersz z pliku dziennika Zabbix agent: „ 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).” Rozpoczyna się on sześcioma pozycjami znaków zawierającymi PID, po których następują data, czas i pozostała część komunikatu. Format czasu dziennika dla tego wiersza to „pppppp:yyyyMMdd:hhmmss”. Należy pamiętać, że znaki „p” i „:” są symbolami zastępczymi i mogą być dowolnymi znakami z wyjątkiem „yMdhms”.
Wypełnia pole inwentaryzacji hosta	Możesz wybrać pole inwentaryzacji hosta, które będzie wypełniane wartością pozycji. Będzie to działać, jeśli dla hosta włączono automatyczne wypełnianie inwentaryzacji oraz w przypadku pozycji typu *Liczbowe (bez znaku)*, *Liczbowe (zmiennoprzecinkowe)*, *Znakowe* i *Tekstowe*.
Opis	Wprowadź opis pozycji. Obsługiwane są makra użytkownika.
Włączona	Zaznacz pole wyboru, aby włączyć pozycję i umożliwić jej przetwarzanie. Po wyłączeniu pozycji jest ona natychmiast usuwana z pamięci podręcznej historii (z wyjątkiem jej ostatniej wartości, która jest zachowywana dla dzienników).
Najnowsze dane	Kliknij łącze, aby wyświetlić najnowsze dane pozycji. To łącze jest dostępne tylko podczas edycji już istniejącej pozycji.

Note:
Pola specyficzne dla typów pozycji opisano na odpowiednich stronach.

Note:
Podczas edycji istniejącej pozycji na poziomie szablonu na poziomie hosta niektóre pola są tylko do odczytu. Możesz użyć łącza w nagłówku formularza, przejść na poziom szablonu i edytować je tam, pamiętając, że zmiany na poziomie szablonu zmienią pozycję dla wszystkich hostów, z którymi szablon jest połączony.

Karta **Znaczniki** umożliwia zdefiniowanie znaczników na poziomie pozycji.



Wstępne przetwarzanie wartości pozycji

Karta **Wstępne przetwarzanie** umożliwia zdefiniowanie reguł transformacji dla odebranych wartości.

Testowanie pozycji

Attention:
Aby przeprowadzić testowanie pozycji, upewnij się, że czas systemowy na serwerze i proxy jest zsynchronizowany. W przypadku gdy czas serwera jest opóźniony, testowanie pozycji może zwrócić komunikat o błędzie ”The task has been expired.” Ustawienie różnych stref czasowych na serwerze i proxy nie wpływa jednak na wynik testu.

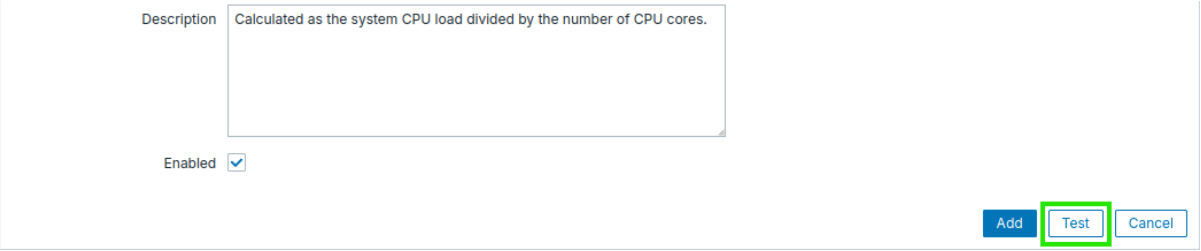
Możliwe jest przetestowanie pozycji i, jeśli jest poprawnie skonfigurowana, uzyskanie w odpowiedzi rzeczywistej wartości. Testowanie może odbyć się nawet przed zapisaniem pozycji.

Testowanie jest dostępne dla pozycji hosta i szablonu, prototypów pozycji oraz reguł niskopoziomowego wykrywania. Testowanie nie jest dostępne dla pozycji aktywnych.

Testowanie pozycji jest dostępne dla następujących pasywnych typów pozycji:

- 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

Aby przetestować pozycję, kliknij przycisk *Test* u dołu formularza konfiguracji pozycji. Zwróć uwagę, że przycisk *Test* będzie wyłączony dla pozycji, których nie można testować (takich jak aktywne sprawdzenia, wykluczone proste sprawdzenia).

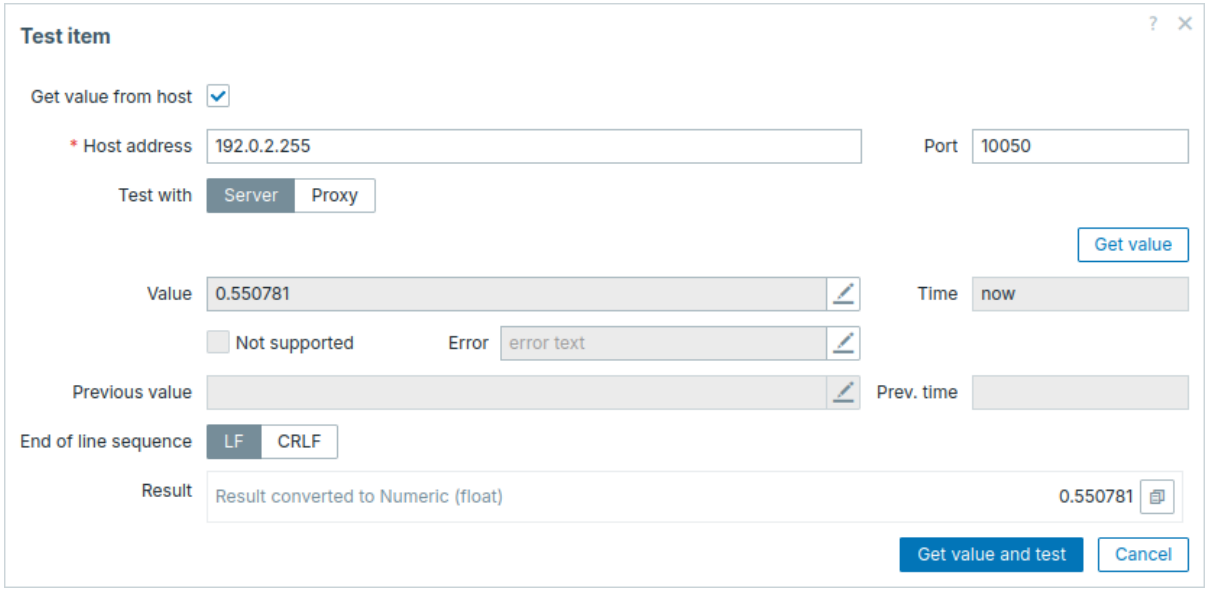


Formularz testowania pozycji zawiera pola dla wymaganych parametrów hosta (adres hosta, port, test z serwerem/proxy (nazwa proxy)) oraz szczegóły specyficzne dla pozycji (takie jak społeczność SNMPv2 lub dane uwierzytelniające bezpieczeństwa SNMPv3). Pola te są zależne od kontekstu:

- Wartości są wstępnie wypełniane, gdy to możliwe, tj. w przypadku pozycji wymagających agenta, na podstawie informacji z wybranego interfejsu agenta hosta.
- Wartości muszą zostać wprowadzone ręcznie dla pozycji szablonu.
- Wartości makr w postaci zwykłego tekstu są rozwijane.
- Pola, w których wartość (lub jej część) jest sekretem lub makrem Vault, są puste i muszą zostać wprowadzone ręcznie. Jeśli dowolny parametr pozycji zawiera wartość makra sekretnego, wyświetlany jest następujący komunikat ostrzegawczy: ”Item contains user-defined macros with secret values. Values of these macros should be entered manually.”
- Pola są wyłączone, gdy nie są potrzebne w kontekście typu pozycji (np. pole adresu hosta i pole proxy są wyłączone dla pozycji obliczanych)

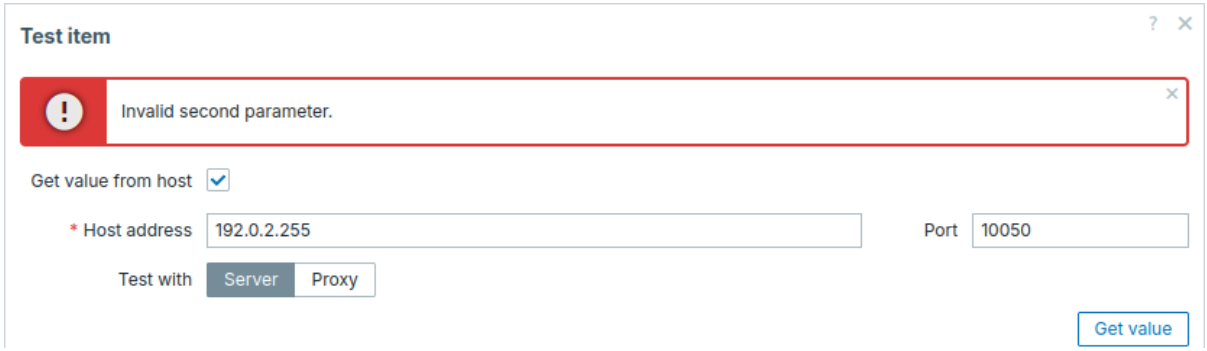
Aby przetestować pozycję, kliknij *Get value*. Jeśli wartość zostanie pobrana pomyślnie, zostanie wpisana do pola *Value*, a bieżąca wartość (jeśli istnieje) zostanie przeniesiona do pola *Previous value*; jednocześnie zostanie obliczone pole *Prev. time*, tj. różnica czasu między dwiema wartościami (kliknięciami), oraz zostanie podjęta próba wykrycia sekwencji EOL i przełączenia na CRLF, jeśli w pobranej wartości zostanie wykryte ”\n\r”.

Kliknij *Get value and test*, aby przetestować preprocessing.



Wartości pobrane z hosta i wyniki testów są przycinane do maksymalnego rozmiaru 512 KB przed wysłaniem do frontend. Jeśli wynik testu zostanie przycięty, wyświetlana jest ikona ostrzeżenia; najedź na nią, aby zobaczyć szczegóły. Jeśli wartość zostanie przycięta, może nie przejść walidacji typu danych w kolejnych testach porównujących duże wartości (np. JSON); w takich przypadkach wyświetlany jest komunikat o błędzie. Zwróć uwagę, że dane większe niż 512 KB są nadal w pełni przetwarzane przez serwer Zabbix.

Jeśli konfiguracja jest nieprawidłowa, wyświetlany jest komunikat o błędzie opisujący możliwą przyczynę.



Pomyślnie pobrana wartość z hosta może być również użyta do testowania kroków preprocessing.

Przyciski formularza

Przyciski u dołu formularza umożliwiają wykonanie kilku operacji.

Add	Dodaj pozycję. Ten przycisk jest dostępny tylko dla nowych pozycji.
Update	Zaktualizuj właściwości pozycji.
Clone	Utwórz kolejną pozycję na podstawie właściwości bieżącej pozycji.
Execute now	Natychmiast wykonaj sprawdzenie nowej wartości pozycji. Obsługiwane tylko w przypadku sprawdzeń **pasywnych** (zobacz więcej szczegółów). *Uwaga*: podczas natychmiastowego sprawdzania wartości pamięć podręczna konfiguracji nie jest aktualizowana, dlatego wartość nie będzie odzwierciedlać najnowszych zmian w konfiguracji pozycji.
Test	Sprawdź, czy konfiguracja pozycji jest prawidłowa, pobierając wartość.
Clear history and trends	Usuń historię i trendy pozycji.
Delete	Usuń pozycję.
Cancel	Anuluj edycję właściwości pozycji.

Limity danych pozycji

Limity danych pozycji zależą od typu danych pozycji oraz backendu bazy danych.

Wartości numeryczne (bez znaku) są przechowywane bez zmian. Jeśli zostanie odebrana wartość zmiennoprzecinkowa, zapisywana jest tylko część całkowita (np. 1.23 → 1).

Wartości numeryczne (float) są przechowywane bez zmian i obsługują około 15 do 17 cyfr precyzji, w zakresie około -1.79E+308 do 1.79E+308. Obsługiwane są również wartości w notacji naukowej (np. 1.23E+7, 1e308, 1.1E-4).

Wartości tekstowe są obcinane przed zapisaniem, aby odpowiadały limitowi typu wartości w bazie danych:

Database	Type of information		
	Character	**Log**	**Text**
MySQL	255 characters	65536 bytes	65536 bytes
PostgreSQL	255 characters	65536 characters	65536 characters
SQLite (only Zabbix proxy)	255 characters	65536 characters	65536 characters

Wartości binarne są ograniczone do 16 MiB (16777216 bajtów). Wartości przekraczające ten limit są odrzucane, a odpowiedni błąd jest wyświetlany we frontendzie.

Wartości typu JSON są ograniczone do 128 MiB (134217728 bajtów). Wartości przekraczające ten limit są odrzucane, a odpowiedni błąd jest wyświetlany we frontendzie. W przypadku MySQL/MariaDB warto rozważyć skonfigurowanie dziennych partycji dla tabeli `history_json` (używanej do przechowywania wartości JSON), ponieważ może ona szybko się rozrastać i blokować housekeeper na dłuższy czas. Jeśli Twój przypadek użycia obejmuje duże wartości JSON (1 MiB lub więcej), zobacz, jak skonfigurować system pod kątem obsługi dużych wartości JSON.

Gdy Zabbix zapisuje dowolne dane pozycji do bazy danych, używa zapytań INSERT i rejestruje je (jeśli `DebugLevel` ma ustawioną wartość 4 lub 5). Duże wpisy dziennika są obcinane do 64 KB.

Limit skryptu niestandardowego

Dostępna długość skryptu niestandardowego zależy od używanej bazy danych:

Baza danych	Limit znaków	Limit bajtów
MySQL	65535	65535
PostgreSQL	65535	bez ograniczeń
SQLite (tylko Zabbix proxy)	65535	bez ograniczeń

Limit czasu pozycji

Limit czasu pozycji określa, jak długo Zabbix powinien czekać przed przerwaniem sprawdzania jako nieudanego.

Gdy zostanie osiągnięty limit czasu, sprawdzanie jest przerywane nawet wtedy, gdy pobieranie danych nie zostało zakończone. Jeśli dane zostaną odebrane częściowo, pozycja staje się niewspierana, a w dzienniku jest zapisywany komunikat o błędzie (na przykład, jeśli dane zostały pomyślnie pobrane tylko dla jednego z wielu OID-ów w sprawdzaniu SNMP).

W przypadku wielu typów pozycji można ustawić **elastyczne** limity czasu pozycji:

- dla pojedynczej pozycji
- dla typu pozycji (na poziomie proxy)
- dla typu pozycji (na poziomie globalnym)

Niestandardowy limit czasu *dla pojedynczej pozycji* jest przydatny, jeśli chcesz określić dłuższy limit czasu dla konkretnej pozycji, zachowując jednocześnie niskie limity czasu dla pozostałych pozycji.

Priorytet limitów czasu

1. Limit czasu pojedynczej pozycji ma pierwszeństwo przed innymi limitami czasu.
2. Limity czasu na poziomie proxy mają pierwszeństwo przed globalnymi limitami czasu.

Limit czasu z konfiguracji serwer Zabbix lub proxy **nie** ma żadnego wpływu na sprawdzania, dla których ustawiono elastyczne limity czasu.

Obsługa elastycznego limitu czasu

Elastyczne limity czasu dla pozycji są obsługiwane dla następujących typów pozycji:

- Zabbix agent (zarówno dla kontroli pasywnych, jak i aktywnych)
- SNMP agent (z wyjątkiem starszych kontroli SNMP[1])
- Simple check (z wyjątkiem `icmpping*` i pozycji VMware[2])
- SSH agent
- Telnet agent
- External check
- Database monitor

- Script
- HTTP agent
- Browser

[1] W przypadku starszych kontroli SNMP (odpytywanie pojedynczego OID) obowiązują ustawienia limitu czasu z konfiguracji serwer lub proxy.
 [2] Dla pozycji `icmpping*` wartość limitu czasu jest określana bezpośrednio w kluczu pozycji; obowiązuje również zakodowany na stałe maksymalny możliwy limit czasu wynoszący 600 sekund. Pozycje monitorowania VMware używają własnego parametru VMwareTimeout z konfiguracji serwer lub proxy.

Nieobsługiwane pozycje

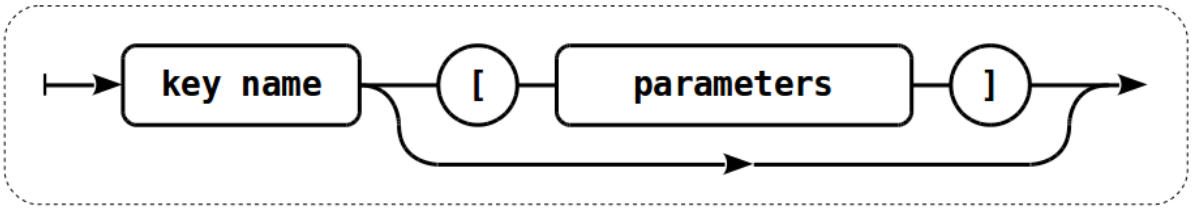
Nieobsługiwane pozycje są zgłaszane ze statusem *Nieobsługiwane* i nadal są ponownie sprawdzane zgodnie z ich standardowym *Interwałem aktualizacji*.

Pozycja staje się nieobsługiwana, jeśli z jakiegoś powodu nie można pobrać jej wartości (np. błąd połączenia, brak skonfigurowanych pollerów do przetwarzania pozycji). Dodatkowo pozycje, które nie otrzymują danych historycznych, pozostają w statusie *Nieobsługiwane*. Jest to oczekiwane zachowanie — pozycja przechodzi do statusu *Normalny* dopiero wtedy, gdy otrzyma nowe, prawidłowe dane historyczne.

Podobnie jak problemy, nieobsługiwane pozycje są ponownie oceniane tylko wtedy, gdy nadejdą nowe dane — nawet jeśli dla danej pozycji nie są już dostępne żadne dane historyczne. Innymi słowy, pozycje i wyzwalacze zmieniają swój stan wyłącznie po otrzymaniu nowych danych. Jednak jeśli wyrażenie wyzwalacza zawiera funkcję date and time i/lub nodata(), wyzwalacz będzie przeliczany co 30 sekund (szczegóły znajdują się w czasie obliczania wyzwalaczy).

Format klucza pozycji

Format klucza pozycji, w tym parametry klucza, musi być zgodny z regułami składni. Poniższe ilustracje przedstawiają obsługiwaną składnię. Dozwolone elementy i znaki w każdym punkcie można określić, podążając za strzałkami - jeśli jakiś blok można osiągnąć przez linię, jest on dozwolony, jeśli nie - nie jest dozwolony.



Aby skonstruować prawidłowy klucz pozycji, należy najpierw określić nazwę klucza, a następnie wybrać, czy mają być podane parametry, czy nie - jak pokazują dwie linie, którymi można podążyć.

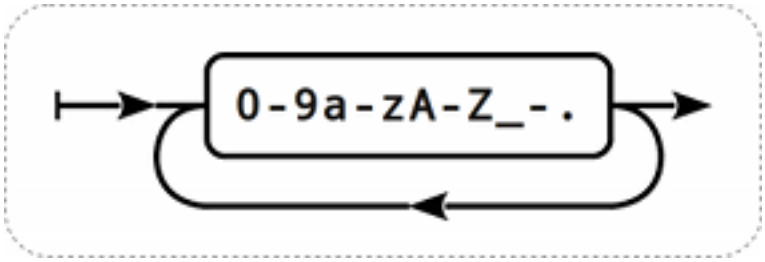
Nazwa klucza

Sama nazwa klucza ma ograniczony zakres dozwolonych znaków, które mogą występować jeden po drugim. Dozwolone znaki to:

`0-9a-zA-Z_-.`

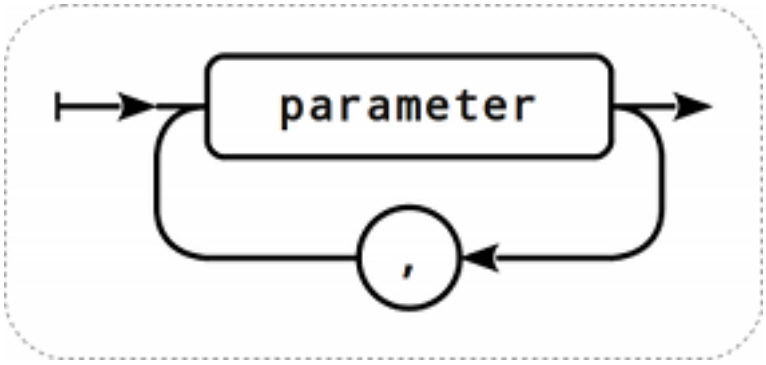
Co oznacza:

- wszystkie cyfry;
- wszystkie małe litery;
- wszystkie wielkie litery;
- podkreślenie;
- myślnik;
- kropka.

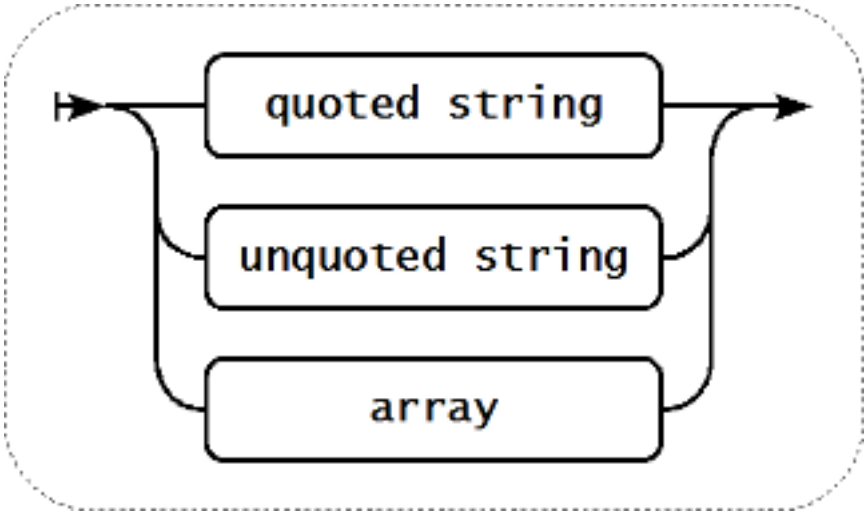


Kluczowe parametry

Klucz pozycji może mieć wiele parametrów oddzielonych przecinkami.



Każdy parametr klucza może być ujętym w cudzysłów ciągiem znaków, ciągiem znaków bez cudzysłowów albo tablicą.

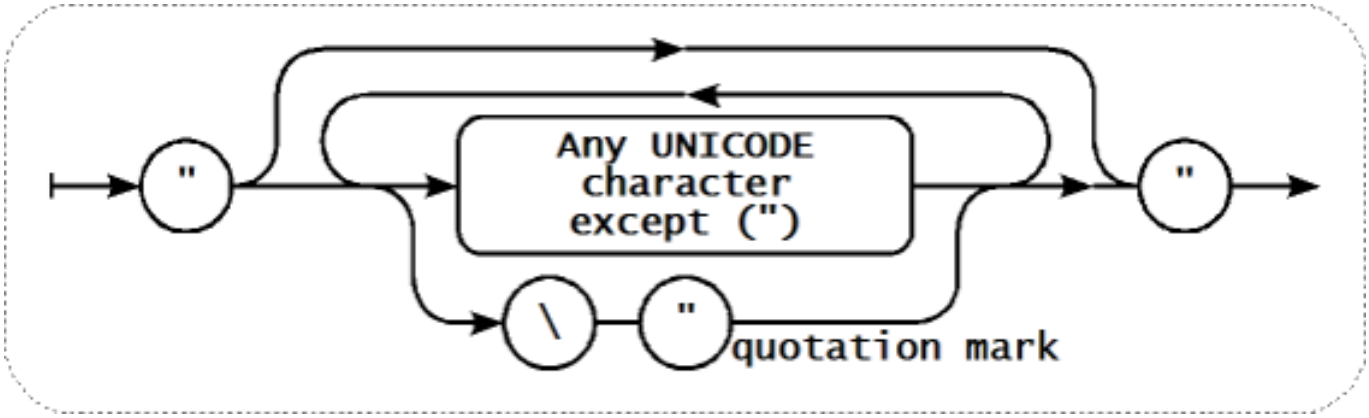


Parametr można również pozostawić pusty, używając w ten sposób wartości domyślnej. W takim przypadku należy dodać odpowiednią liczbę przecinków, jeśli określono jakiekolwiek dalsze parametry. Na przykład klucz pozycji **icmpping[„200„500]** oznacza, że interwał między poszczególnymi pingami wynosi 200 milisekund, limit czasu - 500 milisekund, a wszystkie pozostałe parametry pozostają ustawione na wartości domyślne.

Możliwe jest użycie makr w parametrach. Mogą to być makra użytkownika lub niektóre z wbudowanych makr. Aby sprawdzić, które konkretne wbudowane makra są obsługiwane w parametrach klucza pozycji, wyszukaj na stronie Obsługiwane makra frazę „item key parameters”.

Parametr - ciąg znaków w cudzysłowie

Jeśli parametr klucza jest ciągiem znaków ujętym w cudzysłów, dozwolony jest dowolny znak Unicode. Jeśli ciąg parametru klucza zawiera znak cudzysłowu, parametr ten musi być ujęty w cudzysłów, a każdy cudzysłów będący częścią ciągu parametru musi zostać poprzedzony znakiem ukośnika odwrotnego (\). Jeśli ciąg parametru klucza zawiera przecinek, parametr ten musi być ujęty w cudzysłów.

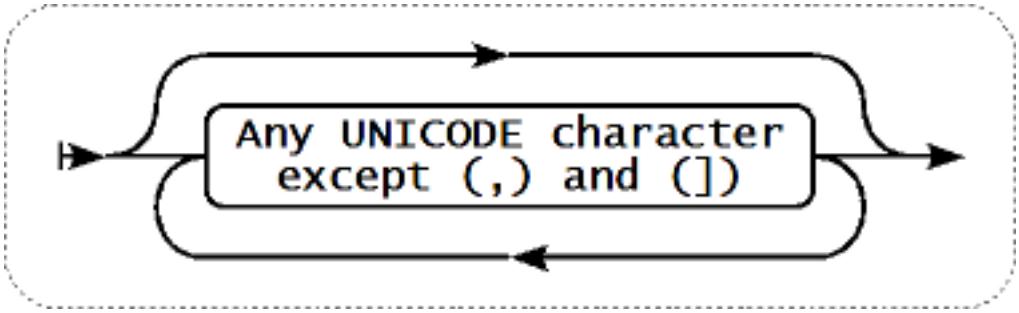


Warning:
Do ujmowania parametrów klucza pozycja w cudzysłów używaj wyłącznie podwójnych cudzysłowów. Pojedyncze cudzysłowy nie są obsługiwane.

Attention:
Wielopoziomowe tablice parametrów, np. `[a,[b,[c,d]],e]`, nie są dozwolone.

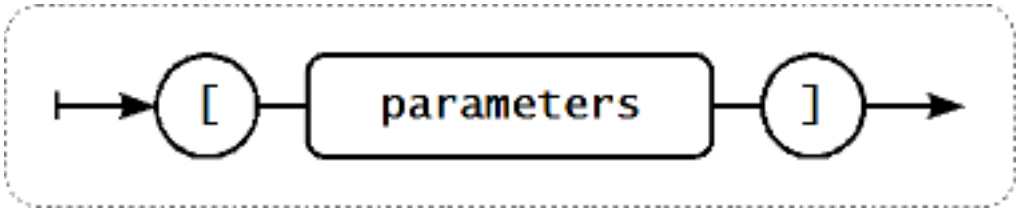
Parametr - ciąg znaków bez cudzysłowu

Jeśli parametr klucza jest ciągiem znaków bez cudzysłowu, dozwolony jest dowolny znak Unicode z wyjątkiem przecinka i prawego nawiasu kwadratowego (]). Parametr bez cudzysłowu nie może zaczynać się od lewego nawiasu kwadratowego ([).



Parametr - array

Jeśli parametr klucza jest tablicą, jest on ponownie ujmowany w nawiasy kwadratowe, gdzie poszczególne parametry są zgodne z zasadami i składnią określania wielu parametrów.



Niestandardowe interwały

Przegląd

Możliwe jest tworzenie niestandardowych reguł dotyczących czasu sprawdzania pozycji. Służą do tego dwie metody: *Elastyczne interwały*, które umożliwiają ponowne zdefiniowanie domyślnego interwału aktualizacji, oraz *Harmonogramowanie*, dzięki któremu sprawdzanie pozycji może być wykonywane o określonej porze lub w określonej sekwencji czasów.

Elastyczne interwały

Elastyczne interwały umożliwiają ponowne zdefiniowanie domyślnego interwału aktualizacji dla określonych przedziałów czasu. Elastyczny interwał jest definiowany za pomocą pól *Interval* i *Period*, gdzie:

- *Interval* - interwał aktualizacji dla określonego przedziału czasu. Obsługiwane są przyrostki czasu, np. 30s, 1m, 2h, 1d.
- *Period* - przedział czasu, w którym elastyczny interwał jest aktywny (szczegółowy opis formatu *Period* znajduje się w sekcji przedziały czasu)

Jeśli wiele elastycznych interwałów nakłada się na siebie, dla nakładającego się okresu używana jest najmniejsza wartość *Interval*. Należy pamiętać, że jeśli najmniejsza wartość nakładających się elastycznych interwałów wynosi „0”, odpytywanie nie będzie wykonywane. Poza elastycznymi interwałami używany jest domyślny interwał aktualizacji.

Należy pamiętać, że jeśli elastyczny interwał jest równy długości okresu, pozycja zostanie sprawdzona dokładnie raz. Jeśli elastyczny interwał jest dłuższy niż okres, pozycja może zostać sprawdzona raz albo nie zostać sprawdzona wcale (dlatego taka konfiguracja nie jest zalecana). Jeśli elastyczny interwał jest krótszy niż okres, pozycja zostanie sprawdzona co najmniej raz.

Jeśli elastyczny interwał jest ustawiony na „0”, pozycja nie jest odpytywana w okresie elastycznego interwału i po zakończeniu tego okresu wznawia odpytywanie zgodnie z domyślnym *Update interval*. Przykłady:

Interval	Period	Description
10	1-5,09:00-18:00	Pozycja będzie sprawdzana co 10 sekund w godzinach pracy.
0	1-7,00:00-7:00	Pozycja nie będzie sprawdzana w nocy.
0	7-7,00:00-24:00	Pozycja nie będzie sprawdzana w niedziele.

Interval	Period	Description
60	1-7,12:00-12:01	Pozycja będzie sprawdzana codziennie o 12:00. Należy pamiętać, że było to stosowane jako obejście dla zaplanowanych kontroli; w przypadku takich kontroli zaleca się używanie interwałów harmonogramu.

Interwały harmonogramu

Interwały harmonogramu są używane do sprawdzania pozycji o określonych porach. Podczas gdy elastyczne interwały służą do redefiniowania domyślnego interwału aktualizacji pozycji, interwały harmonogramu są używane do określenia niezależnego harmonogramu sprawdzania, który jest wykonywany równolegle.

Interwał harmonogramu jest definiowany jako: `md<filter>wd<filter>h<filter>m<filter>s<filter>`, gdzie:

- **md** - dni miesiąca
- **wd** - dni tygodnia
- **h** - godziny
- **m** - minuty
- **s** - sekundy

`<filter>` służy do określania wartości dla swojego prefiksu (dni, godziny, minuty, sekundy) i jest definiowany jako: `[<from>[-<to>]][/<step>][,<filter>]`, gdzie:

- `<from>` i `<to>` definiują zakres pasujących wartości (włącznie). Jeśli `<to>` zostanie pominięte, filtr dopasowuje zakres `<from> - <from>`. Jeśli `<from>` również zostanie pominięte, filtr dopasowuje wszystkie możliwe wartości.
- `<step>` definiuje krok przeskoku wartości liczbowych w obrębie zakresu. Domyślnie `<step>` ma wartość 1, co oznacza, że dopasowywane są wszystkie wartości zdefiniowanego zakresu.

Chociaż definicje filtrów są opcjonalne, musi zostać użyty co najmniej jeden filtr. Filtr musi mieć zdefiniowany albo zakres, albo wartość *<step>*.

Pusty filtr dopasowuje albo „0”, jeśli nie zdefiniowano filtra niższego poziomu, albo wszystkie możliwe wartości w przeciwnym razie. Na przykład, jeśli filtr godzin zostanie pominięty, dopasowana zostanie tylko godzina „0”, pod warunkiem że filtry minut i sekund również są pominięte; w przeciwnym razie pusty filtr godzin będzie dopasowywał wszystkie wartości godzin.

Prawidłowe wartości `<from>` i `<to>` dla odpowiednich prefiksów filtrów to:

Prefix	Opis	*<from>*	*<to>*
md	Dni miesiąca	1-31	1-31
wd	Dni tygodnia	1-7	1-7
h	Godziny	0-23	0-23
m	Minuty	0-59	0-59
s	Sekundy	0-59	0-59

Wartość `<from>` musi być mniejsza lub równa wartości `<to>`. Wartość `<step>` musi być większa lub równa 1 oraz mniejsza lub równa `<to>` - `<from>`.

Jednocyfrowe wartości dni miesiąca, godzin, minut i sekund mogą być poprzedzone zerem. Na przykład `md01-31` i `h/02` są prawidłowymi interwałami, ale `md01-031` i `wd01-07` już nie.

W Zabbix frontend wiele interwałów harmonogramu wprowadza się w osobnych wierszach. W Zabbix API są one łączone w jeden ciąg znaków z użyciem średnika `;` jako separatora.

Jeśli dany czas pasuje do kilku interwałów, zostanie wykonany tylko raz. Na przykład `wd1h9;h9` zostanie wykonane tylko raz w poniedziałek o 9:00.

Przykłady:

Interwał	Zostanie wykonane
m0-59	co minutę
h9-17/2	co 2 godziny, zaczynając od 9:00 (9:00, 11:00 ...)
m0,30 or m/30	co godzinę o hh:00 i hh:30
m0,5,10,15,20,25,30,35,40,45,50,55 or m/5	co pięć minut
wd1-5h9	w każdy poniedziałek do piątku o 9:00

Interwał	Zostanie wykonane
wd1-5h9-18	w każdy poniedziałek do piątku o 9:00,10:00,...,18:00
h9,10,11 or h9-11	codziennie o 9:00, 10:00 i 11:00
md1h9m30	każdego 1. dnia miesiąca o 9:30
md1wd1h9m30	każdego 1. dnia miesiąca o 9:30, jeśli jest to poniedziałek
h9m/30	codziennie o 9:00, 9:30
h9m0-59/30	codziennie o 9:00, 9:30
h9,10m/30	codziennie o 9:00, 9:30, 10:00, 10:30
h9-10m30	codziennie o 9:30, 10:30
h9m10-40/30	codziennie o 9:10, 9:40
h9,10m10-40/30	codziennie o 9:10, 9:40, 10:10, 10:40
h9-10m10-40/30	codziennie o 9:10, 9:40, 10:10, 10:40
h9m10-40	codziennie o 9:10, 9:11, 9:12, ... 9:40
h9m10-40/1	codziennie o 9:10, 9:11, 9:12, ... 9:40
h9-12,15	codziennie o 9:00, 10:00, 11:00, 12:00, 15:00
h9-12,15m0	codziennie o 9:00, 10:00, 11:00, 12:00, 15:00
h9-12,15m0s30	codziennie o 9:00:30, 10:00:30, 11:00:30, 12:00:30, 15:00:30
h9-12s30	codziennie o 9:00:30, 9:01:30, 9:02:30 ... 12:58:30, 12:59:30
h9m/30;h10 (*składnia specyficzna dla API*)	codziennie o 9:00, 9:30, 10:00
h9m/30 h10 (*dodaj to jako kolejny wiersz w frontend*)	codziennie o 9:00, 9:30, 10:00

Wyrównanie stref czasowych dla proxy i agent

Należy pamiętać, że proxy i agent Zabbix używają swoich lokalnych stref czasowych podczas przetwarzania interwałów harmonogramu.

Z tego powodu, gdy interwały harmonogramu są stosowane do pozycji monitorowanych przez proxy Zabbix lub aktywnych pozycji agent, zaleca się ustawienie strefy czasowej odpowiednich proxy lub agent na taką samą jak na serwerze Zabbix, w przeciwnym razie kolejka może nieprawidłowo zgłaszać opóźnienia pozycji.

Strefę czasową dla proxy Zabbix lub agent można ustawić za pomocą zmiennej środowiskowej TZ w pliku jednostki `systemd`:

```
[Service]
...
Environment="TZ=Europe/Amsterdam"
```

Wstępne przetwarzanie wartości pozycji

Przegląd

Wstępne przetwarzanie umożliwia stosowanie przekształceń do odebranych wartości pozycji przed zapisaniem ich do bazy danych. Te przekształcenia/kroki wstępnego przetwarzania są wykonywane przez serwer Zabbix lub proxy (jeśli pozycje są monitorowane przez proxy).

Ta funkcja obsługuje różne przypadki użycia, takie jak:

- konwersja bajtów na bity (np. mnożenie wartości ruchu sieciowego przez „8”);
- obliczanie statystyk na sekundę dla wartości narastających przyrostowo;
- stosowanie wyrażeń regularnych do wyodrębniania lub modyfikowania wartości;
- wykonywanie własnych skryptów na wartościach;
- odrzucanie niezmienionych wartości w celu optymalizacji przechowywania w bazie danych.

Dla pozycji można skonfigurować jeden lub więcej kroków wstępnego przetwarzania. Kroki te są wykonywane w kolejności, w jakiej zostały skonfigurowane.

Attention:
Jeśli krok wstępnego przetwarzania zakończy się niepowodzeniem, pozycja staje się nieobsługiwana. Można tego uniknąć dzięki obsłudze błędów *Custom on fail* (dostępnej dla większości przekształceń), która pozwala odrzucać wartości lub ustawiać wartości niestandardowe.

 W przypadku pozycji logów metadane logu (bez wartości) zawsze resetują stan nieobsługiwanej pozycji, ponownie czyniąc ją obsługiwaną. Dzieje się tak nawet wtedy, gdy początkowy błąd wystąpił po odebraniu wartości logu od agent.

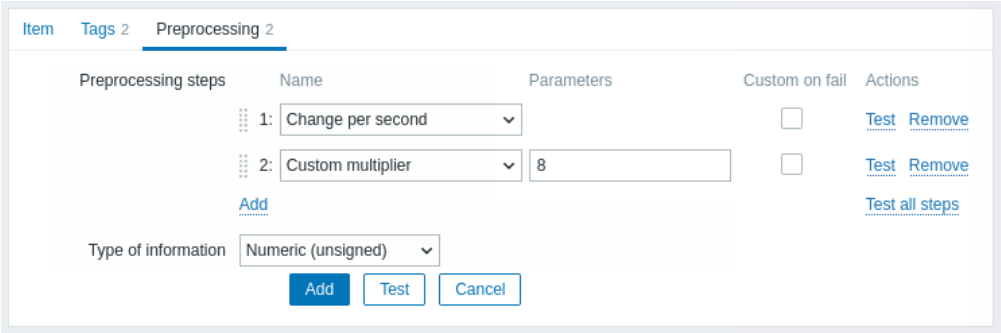
Wszystkie wartości przekazywane do wstępnego przetwarzania są początkowo traktowane jako ciągi znaków. Konwersja do żądanego typu wartości (zgodnie z definicją w konfiguracji pozycji) jest wykonywana na końcu potoku wstępnego przetwarzania. Jednak określone kroki wstępnego przetwarzania mogą wymuszać wcześniejsze konwersje, jeśli jest to wymagane. Szczegółowe informacje techniczne można znaleźć w sekcji Szczegóły wstępnego przetwarzania.

Aby upewnić się, że konfiguracja wstępnego przetwarzania działa zgodnie z oczekiwaniami, możesz ją przetestować.

Zobacz także: Przykłady wstępnego przetwarzania

Konfiguracja

Kroki przetwarzania wstępnego definiuje się na karcie **Przetwarzanie wstępne** w formularzu konfiguracji pozycji.



Kliknij *Dodaj*, aby wybrać obsługiwaną transformację.

Pole *Typ informacji* jest wyświetlane u dołu karty, gdy zdefiniowano co najmniej jeden krok przetwarzania wstępnego. W razie potrzeby można zmienić typ informacji bez opuszczania karty *Przetwarzanie wstępne*. Szczegółowy opis parametrów znajduje się w sekcji Tworzenie pozycji.

Obsługiwane transformacje

Wszystkie obsługiwane transformacje wymieniono poniżej.
Kliknij nazwę transformacji, aby wyświetlić jej pełne szczegóły.

Nazwa	Opis	Typ
Wyrażenie regularne	Dopasowanie wartości do wyrażenia regularnego i zastąpienie jej wymaganym wynikiem.	Tekst
Zastąp	Znalezienie wyszukiwanego ciągu i zastąpienie go innym ciągiem (lub niczym).	
Przytnij	Usunięcie określonych znaków z początku i końca wartości.	
Przytnij z prawej	Usunięcie określonych znaków z końca wartości.	
Przytnij z lewej	Usunięcie określonych znaków z początku wartości.	
XML XPath	Wyodrębnienie wartości lub fragmentu z danych XML za pomocą funkcji XPath.	Dane strukturalne
JSON Path	Wyodrębnienie wartości lub fragmentu z danych JSON za pomocą funkcji JSONPath.	
CSV do JSON	Konwersja danych pliku CSV do formatu JSON.	
XML do JSON	Konwersja danych w formacie XML do formatu JSON.	
Wartość SNMP walk	Wyodrębnienie wartości według określonej nazwy OID/MIB i zastosowanie opcji formatowania.	SNMP
SNMP walk do JSON	Konwersja wartości SNMP do formatu JSON.	
Pobierz wartość SNMP	Zastosowanie opcji formatowania do wartości pobranej za pomocą SNMP get.	
Niestandardowy mnożnik	Pomnożenie wartości przez określoną liczbę całkowitą lub zmiennoprzecinkową.	Arytmetyka
Prosta zmiana	Obliczenie różnicy między bieżącą a poprzednią wartością.	Zmiana
Zmiana na sekundę	Obliczenie szybkości zmiany wartości (różnicy między bieżącą a poprzednią wartością) na sekundę.	
Wartość logiczna na dziesiętną	Konwersja wartości z formatu logicznego do dziesiętnego.	Systemy liczbowe
Ósemkowa na dziesiętną	Konwersja wartości z formatu ósemkowego do dziesiętnego.	
Szesnastkowa na dziesiętną	Konwersja wartości z formatu szesnastkowego do dziesiętnego.	
JavaScript	Wprowadzenie kodu JavaScript.	Skrypty niestandardowe
W zakresie	Zdefiniowanie zakresu, w którym powinna znajdować się wartość.	Walidacja

Nazwa	Opis	Typ
Pasuje do wyrażenia regularnego	Określenie wyrażenia regularnego, do którego wartość musi pasować.	
Nie pasuje do wyrażenia regularnego	Określenie wyrażenia regularnego, do którego wartość nie może pasować.	
Sprawdź błąd w JSON	Sprawdzenie komunikatu o błędzie na poziomie aplikacji znajdującego się pod adresem JSONPath.	
Sprawdź błąd w XML	Sprawdzenie komunikatu o błędzie na poziomie aplikacji znajdującego się pod adresem XPath.	
Sprawdź błąd za pomocą wyrażenia regularnego	Sprawdzenie komunikatu o błędzie na poziomie aplikacji za pomocą wyrażenia regularnego.	
Sprawdź, czy wartość jest nieobsługiwana	Sprawdzenie, czy nie można było pobrać wartości pozycji.	
Odrzuć niezmienione	Odrzucenie wartości, jeśli nie uległa zmianie.	Ograniczanie
Odrzuć niezmienione z sygnałem kontrolnym	Odrzucenie wartości, jeśli nie uległa zmianie w określonym czasie.	
Wzorzec Prometheus	Użycie następującego zapytania do wyodrębnienia wymaganych danych z metryk Prometheus.	Prometheus
Prometheus do JSON	Konwersja wymaganych metryk Prometheus do formatu JSON.	

Jeśli dla wybranej transformacji włączono pole wyboru *Niestandardowe działanie w przypadku błędu*, można je zaznaczyć i określić niestandardowe opcje obsługi błędów: *Odrzuć wartość* (odrzucenie wartości), *Ustaw wartość na* (ustawienie określonej wartości) lub *Ustaw błąd na* (ustawienie określonego komunikatu o błędzie).
Jeśli krok wstępnego przetwarzania zakończy się niepowodzeniem, pozycja nie otrzyma statusu nieobsługiwanej, gdy wybrano opcję *Odrzuć wartość* lub *Ustaw wartość na*.

Należy pamiętać, że w przypadku kroków wstępnego przetwarzania *Zmiana* i *Ograniczanie* Zabbix musi zapamiętać ostatnią wartość, aby obliczyć lub porównać nową wartość zgodnie z wymaganiami.
Te poprzednie wartości są obsługiwane przez menedżera wstępnego przetwarzania.
Po ponownym uruchomieniu serwera Zabbix lub proxy albo po wprowadzeniu jakiejkolwiek zmiany w krokach wstępnego przetwarzania ostatnia wartość odpowiedniej pozycji jest resetowana, co skutkuje następującymi sytuacjami:

- w przypadku kroków *Prosta zmiana* i *Zmiana na sekundę* następna wartość zostanie zignorowana, ponieważ nie ma poprzedniej wartości, na podstawie której można obliczyć zmianę;
- w przypadku kroków *Odrzuć niezmienione* i *Odrzuć niezmienione z sygnałem kontrolnym* następna wartość nigdy nie zostanie odrzucona, nawet jeśli zgodnie z regułami odrzucania powinna zostać odrzucona.

Wyrażenie regularne

Dopasuj wartość do wyrażenia regularnego i zastąp ją wymaganym wynikiem.

Parametry:

- **pattern** - wyrażenie regularne;
- **output** - szablon formatowania wyniku. Sekwencja escape \N (gdzie N=1...9) jest zastępowana przez N-tą dopasowaną grupę. Sekwencja escape \0 jest zastępowana przez dopasowany tekst.

Uwagi:

- Brak dopasowania wartości wejściowej spowoduje, że pozycja będzie nieobsługiwana;
- Wyrażenie regularne obsługuje wyodrębnianie maksymalnie 10 przechwyconych grup za pomocą sekwencji \N;
- Więcej informacji i przykładów można znaleźć w sekcji wyrażeń regularnych.

Zastąp

Znajdź ciąg wyszukiwania i zastąp go innym ciągiem (lub niczym).

Parametry:

- **ciąg wyszukiwania** - ciąg do znalezienia i zastąpienia, z uwzględnieniem wielkości liter (wymagany);
- **zastąpienie** - ciąg, którym należy zastąpić ciąg wyszukiwania. Ciąg zastępujący może być również pusty, co pozwala skutecznie usunąć znaleziony ciąg wyszukiwania.

Uwagi:

- Wszystkie wystąpienia ciągu wyszukiwania zostaną zastąpione;
- Można używać sekwencji ucieczki do wyszukiwania lub zastępowania znaków nowej linii, powrotu karetki, tabulatorów i spacji ”\n \r \t \s”; ukośnik odwrotny można zapisać jako ”\\” a sekwencje ucieczki można zapisać jako ”\\n”;

- Podczas wykrywania niskopoziomowego znaki nowej linii, powrotu karetki i tabulatorów są automatycznie zapisywane przy użyciu sekwencji ucieczki.

Przytnij

Usuń określone znaki z początku i końca wartości.

Przycinanie z prawej strony

Usuń określone znaki z końca wartości.

Przycinanie z lewej

Usuń określone znaki z początku wartości.

XML XPath

Wyodrębnianie wartości lub fragmentu z danych XML za pomocą funkcji XPath.

Uwagi:

- Aby ta opcja działała, serwer Zabbix (lub proxy Zabbix) musi być skompilowany z obsługą libxml;
- Przestrzenie nazw nie są obsługiwane.

Przykłady:

```
number(/document/item/value) #wyodrębni „10” z <document><item><value>10</value></item></document>
number(/document/item/@attribute) #wyodrębni „10” z <document><item attribute="10"></item></document>
/document/item #wyodrębni „<item><value>10</value></item>” z <document><item><value>10</value></item></doc
```

JSON Path

Wyodrębnij wartość lub fragment z danych JSON za pomocą funkcji JSONPath.

CSV do JSON

Konwertuj dane pliku CSV do formatu JSON.

Więcej informacji znajduje się tutaj: Wstępne przetwarzanie CSV do JSON.

XML do JSON

Konwertuje dane w formacie XML do JSON.

Więcej informacji można znaleźć w sekcji: Reguły serializacji.

Wartość SNMP walk

Wyodrębnij wartość według określonego OID/nazwy MIB i zastosuj opcje formatowania:

- **Bez zmian** - zwraca ciąg szesnastkowy jako niezmieniony ciąg szesnastkowy (*uwaga*: wskazówki dotyczące wyświetlania są nadal stosowane);
- **UTF-8 z hex-STRING** - konwertuje ciąg szesnastkowy na ciąg UTF-8;
- **MAC z hex-STRING** - sprawdza poprawność ciągu szesnastkowego jako adresu MAC i zwraca prawidłowy ciąg adresu MAC (gdzie ' ' są zastępowane przez ':');
- **Liczba całkowita z BITS** - konwertuje pierwsze 8 bajtów ciągu bitów wyrażonego jako sekwencja znaków szesnastkowych (np. „1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W ciągach bitów dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane.

SNMP walk do JSON

Konwertowanie wartości SNMP do formatu JSON.

Określ nazwę pola w formacie JSON oraz odpowiadającą jej ścieżkę OID SNMP. Wartości pól zostaną uzupełnione wartościami z określonej ścieżki OID SNMP.

Uwagi:

- Dostępne są opcje formatowania wartości podobne do tych w kroku *Wartość SNMP walk*;
- Tego kroku przetwarzania wstępnego można użyć do wykrywania OID SNMP.

Pobierz wartość SNMP

Zastosuj opcje formatowania do wartości pobranej za pomocą SNMP:

- **UTF-8 z Hex-STRING** - konwertuje ciąg szesnastkowy na ciąg UTF-8;
- **MAC z Hex-STRING** - sprawdza poprawność ciągu szesnastkowego jako adresu MAC i zwraca prawidłowy ciąg adresu MAC (gdzie ' ' są zastępowane przez ':');

- **Liczba całkowita z BITS** - konwertuje pierwsze 8 bajtów ciągu bitów wyrażonego jako sekwencja znaków szesnastkowych (np. „1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W przypadku ciągów bitów dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane.

Niestandardowy mnożnik

Pomnóż wartość przez określoną liczbę całkowitą lub zmiennoprzecinkową.

Uwagi:

- Użyj tej opcji, aby konwertować wartości otrzymywane w KB, MBps itp. na B, Bps. W przeciwnym razie Zabbix nie będzie mógł poprawnie ustawić prefiksów (K, M, G itd.).
- Pamiętaj, że jeśli typem informacji pozycji jest *Numeric (unsigned)*, przychodzące wartości z częścią ułamkową zostaną obcięte (tzn. „0.9” zmieni się w „0”) przed zastosowaniem niestandardowego mnożnika;
- Jeśli używasz niestandardowego mnożnika lub przechowujesz wartość jako *Change per second* dla pozycji z typem informacji ustawionym na *Numeric (unsigned)*, a wynikowa obliczona wartość jest w rzeczywistości liczbą zmiennoprzecinkową, obliczona wartość nadal zostanie zaakceptowana jako poprawna przez obcięcie części dziesiętnej i zapisanie wartości jako liczby całkowitej;
- Obsługiwane są: notacja naukowa, na przykład 1e+70; makra użytkownika i makra LLD; ciągi zawierające makra, na przykład `{#MACRO}e+10`, `{$MACRO1}e+{$MACRO2}`. Makra muszą zostać rozwinięte do liczby całkowitej lub zmiennoprzecinkowej.

Prosta zmiana

Oblicza różnicę między bieżącą a poprzednią wartością.

Uwagi:

- Ten krok może być przydatny do pomiaru stale rosnącej wartości;
- Obliczana jako **value**-**prev_value**, gdzie *value* - bieżąca wartość; *prev_value* - poprzednio otrzymana wartość;
- Dozwolona jest tylko jedna operacja zmiany na pozycję („Prosta zmiana” lub „Zmiana na sekundę”);
- Jeśli bieżąca wartość jest mniejsza niż poprzednia wartość, Zabbix odrzuca tę różnicę (niczego nie zapisuje) i czeka na kolejną wartość.

Zmiana na sekundę

Oblicza szybkość zmiany wartości (różnicę między bieżącą a poprzednią wartością) na sekundę.

Uwagi:

- Ten krok jest przydatny do obliczania szybkości przyrostu stale rosnącej wartości;
- Ponieważ to obliczenie może zwracać liczby zmiennoprzecinkowe, zaleca się ustawienie opcji „Typ informacji” na *Liczbowe (zmiennoprzecinkowe)*, nawet jeśli przychodzące wartości surowe są liczbami całkowitymi. Ma to szczególne znaczenie w przypadku małych liczb, dla których część dziesiętna ma znaczenie. Jeśli wartości zmiennoprzecinkowe są duże i mogą przekroczyć długość pola „float”, w wyniku czego cała wartość może zostać utracona, zaleca się użycie opcji *Liczbowe (bez znaku)*, aby obcięta została tylko część dziesiętna.
- Obliczane jako (**value**-**prev_value**)/(**time**-**prev_time**), gdzie *value* - bieżąca wartość; *prev_value* - poprzednio odebrana wartość; *time* - bieżący znacznik czasu; *prev_time* - znacznik czasu poprzedniej wartości;
- Dozwolona jest tylko jedna operacja zmiany na pozycję („Prosta zmiana” lub „Zmiana na sekundę”);
- Jeśli bieżąca wartość jest mniejsza niż poprzednia wartość, Zabbix odrzuca tę różnicę (niczego nie zapisuje) i czeka na kolejną wartość. Pomaga to prawidłowo obsługiwać na przykład przepełnienie 32-bitowych liczników SNMP.

Wartość logiczna na dziesiętną

Konwertuje wartość z formatu logicznego na dziesiętny.

Uwagi:

- Reprezentacja tekstowa jest tłumaczona na wartość 0 lub 1. W związku z tym wartość „TRUE” jest zapisywana jako 1, a „FALSE” jako 0. Wszystkie wartości są dopasowywane bez uwzględniania wielkości liter. Obecnie rozpoznawane wartości to: dla *TRUE* - true, t, yes, y, on, up, running, enabled, available, ok, master; dla *FALSE* - false, f, no, n, off, down, unused, disabled, unavailable, err, slave. Ponadto każda niezerowa wartość liczbowa jest uznawana za TRUE, a zero za FALSE.

Ósemkowy na dziesiętny

Konwertuje wartość z formatu ósemkowego na dziesiętny.

Szesnastkowy na dziesiętny

Konwertuje wartość z formatu szesnastkowego na dziesiętny.

JavaScript

Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w polu parametru lub ikonie ołówka obok niego.

Attention:
Nie używaj niezadeklarowanych przypisań w kodzie JavaScript przetwarzania wstępnego. Użyj `var`, aby zadeklarować zmienne lokalne.

Uwagi:

- Dostępna długość kodu JavaScript zależy od używanej bazy danych;
- Więcej informacji można znaleźć tutaj: Przetwarzanie wstępne JavaScript.

W zakresie

Zdefiniuj zakres, w którym powinna znajdować się wartość, określając wartości minimalną/maksymalną (włącznie).

Uwagi:

- Akceptowane są wartości liczbowe (w tym dowolna liczba cyfr, opcjonalna część dziesiętna i opcjonalna część wykładnicza, wartości ujemne);
- Wartość minimalna powinna być mniejsza od maksymalnej;
- Musi istnieć co najmniej jedna wartość;
- Można używać makr użytkownika i makr wykrywania niskiego poziomu.

Dopasowuje wyrażenie regularne

Określ wyrażenie regularne, które musi pasować do wartości.

Nie pasuje do wyrażenia regularnego

Określ wyrażenie regularne, któremu wartość nie może odpowiadać.

Sprawdź, czy w formacie JSON wystąpił błąd

Sprawdź, czy w ścieżce JSONPath znajduje się komunikat o błędzie na poziomie aplikacji. Zatrzymaj przetwarzanie, jeśli operacja zakończyła się powodzeniem, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie z wartością, która obowiązywała przed tym krokiem przetwarzania wstępnego.

Uwagi:

- Te błędy usług zewnętrznych są zgłaszane użytkownikowi w niezmienionej postaci, bez dodawania informacji o kroku przetwarzania wstępnego;
- W przypadku niepowodzenia analizy nieprawidłowego formatu JSON nie zostanie zgłoszony żaden błąd.

Sprawdź, czy w XML wystąpił błąd

Sprawdź, czy w ścieżce XPath znajduje się komunikat o błędzie na poziomie aplikacji. Zatrzymaj przetwarzanie, jeśli operacja zakończyła się powodzeniem, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie z wartością sprzed tego kroku przetwarzania wstępnego.

Uwagi:

- Te błędy usług zewnętrznych są zgłaszane użytkownikowi w niezmienionej postaci, bez dodawania informacji o kroku przetwarzania wstępnego;
- W przypadku niepowodzenia analizy nieprawidłowego XML nie zostanie zgłoszony żaden błąd.

Sprawdź błąd za pomocą wyrażenia regularnego

Sprawdź komunikat o błędzie na poziomie aplikacji za pomocą wyrażenia regularnego.
Zatrzymaj przetwarzanie, jeśli operacja zakończyła się powodzeniem, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie z wartością, która obowiązywała przed tym krokiem wstępnego przetwarzania.

Parametry:

- **pattern** - wyrażenie regularne;
- **output** - szablon formatowania danych wyjściowych.
  Sekwencja ucieczki \N (gdzie N=1…9) jest zastępowana N-tą dopasowaną grupą.
  Sekwencja ucieczki \0 jest zastępowana dopasowanym tekstem.

Uwagi:

- Te błędy usług zewnętrznych są zgłaszane użytkownikowi w niezmienionej postaci, bez dodawania informacji o kroku wstępnego przetwarzania.

Sprawdź, czy wartość jest nieobsługiwana

Sprawdź, czy nie można pobrać wartości pozycji. Określ sposób przetwarzania błędu na podstawie analizy zwróconego komunikatu o błędzie.

Parametry:

- **scope** - wybierz zakres przetwarzania błędów:
any error - dowolny błąd;
error matches - tylko błąd pasujący do wyrażenia regularnego określonego w polu *pattern*;
error does not match - tylko błąd, który nie pasuje do wyrażenia regularnego określonego w polu *pattern*
- **pattern** - wyrażenie regularne, do którego ma pasować błąd. Jeśli w parametrze zakresu wybrano *any error*, to pole nie jest wyświetlane. Jeśli pole jest wyświetlane, jego wypełnienie jest wymagane.

Uwagi:

- Zwykle brak wartości lub niepowodzenie jej pobrania powoduje, że pozycja staje się nieobsługiwana. Ten krok wstępnego przetwarzania umożliwia zmianę tego zachowania.
- Ten krok wstępnego przetwarzania sprawdza tylko, czy nie można pobrać wartości pozycji. Nie sprawdza na przykład, czy typ pobranej wartości (np. ciąg znaków) odpowiada typowi informacji pozycji (np. wartości numerycznej); szczegółowe informacje zawiera sekcja Przykłady wstępnego przetwarzania. W przypadku niezgodności typów pozycja może nadal stać się nieobsługiwana po wykonaniu wszystkich kroków wstępnego przetwarzania. Aby sprawdzić niezgodność typów, można na przykład użyć kroku wstępnego przetwarzania *Custom multiplier*; zobacz Przykłady wstępnego przetwarzania.
- Przechwytywanie grup wyrażenia regularnego jest obsługiwane w polach *Set value to* lub *Set error to*. Użyj \N (gdzie N=1…9), aby pobrać N. dopasowaną grupę; użyj \0, aby pobrać dopasowany tekst;
- Te kroki są zawsze wykonywane jako pierwsze kroki wstępnego przetwarzania i po zapisaniu zmian w pozycji są umieszczane nad wszystkimi pozostałymi krokami;
- Obsługiwanych jest wiele kroków *Check for not supported value*, wykonywanych w określonej kolejności. Krok *any error* zostanie automatycznie umieszczony jako ostatni krok w tej grupie.

Odrzuć niezmienione

Odrzuć wartość, jeśli nie uległa zmianie.

Uwagi:

- Jeśli wartość zostanie odrzucona, nie jest zapisywana w bazie danych, a serwer Zabbix nie ma wiedzy o tym, że ta wartość została odebrana. Wyrażenia wyzwalaczy nie będą oceniane, w rezultacie nie zostaną utworzone/rozwiązane żadne problemy dla powiązanych wyzwalaczy. Funkcje będą działać wyłącznie na podstawie danych, które są faktycznie zapisane w bazie danych. Ponieważ trendy są tworzone na podstawie danych w bazie danych, jeśli przez godzinę nie zostanie zapisana żadna wartość, to dla tej godziny również nie będzie danych trendów.
- Dla jednej pozycji można określić tylko jedną opcję ograniczania.

Odrzucanie niezmienionych wartości z użyciem sygnału heartbeat

Odrzuca wartość, jeśli nie zmieniła się w określonym czasie (w sekundach).

Uwagi:

- Do określenia liczby sekund można używać dodatnich liczb całkowitych (minimum - 1 sekunda);
- Można używać przyrostków czasu (np. `30s`, `1m`, `2h`, `1d`);
- Można używać makr użytkownika i makr niskopoziomowego wykrywania;
- Jeśli wartość zostanie odrzucona, nie zostanie zapisana w bazie danych, a serwer Zabbix nie będzie wiedział, że ta wartość została odebrana. Wyrażenia wyzwalaczy nie będą obliczane, dlatego nie zostaną utworzone ani rozwiązane żadne problemy dotyczące powiązanych wyzwalaczy. Funkcje będą działać wyłącznie na podstawie danych faktycznie zapisanych w bazie danych. Ponieważ trendy są tworzone na podstawie danych w bazie danych, jeśli przez godzinę nie zostanie zapisana żadna wartość, dla tej godziny nie będą również dostępne dane trendów.
- Dla jednej pozycji można określić tylko jedną opcję ograniczania.

Wzorzec Prometheus

Użyj następującego zapytania, aby wyodrębnić wymagane dane z metryk Prometheus.

Więcej informacji znajdziesz w sekcji Sprawdzanie Prometheus.

Prometheus do JSON

Konwertuje wymagane metryki Prometheus do formatu JSON.

Więcej informacji znajdziesz w sekcji Sprawdzanie Prometheus.

Obsługa makr

Makra użytkownika oraz makra użytkownika z kontekstem są obsługiwane w:

- parametrach kroku przetwarzania wstępnego, w tym w kodzie JavaScript;
- parametrach niestandardowej obsługi błędów (pola *Ustaw wartość na* i *Ustaw błąd na*).

Note:
Kontekst makra jest ignorowany, gdy makro jest zastępowane swoją wartością. Wartość makra jest wstawiana do kodu w postaci niezmienionej; nie ma możliwości dodania dodatkowego escapingu przed umieszczeniem wartości w kodzie JavaScript. Należy pamiętać, że w niektórych przypadkach może to powodować błędy JavaScript.

Testowanie

Zobacz testowanie preprocessing.

Testowanie przetwarzania wstępnego

Testowanie

Testowanie kroków przetwarzania wstępnego jest przydatne, aby upewnić się, że złożone potoki przetwarzania wstępnego dają oczekiwane wyniki, bez czekania na odebranie i przetworzenie wartości pozycja.

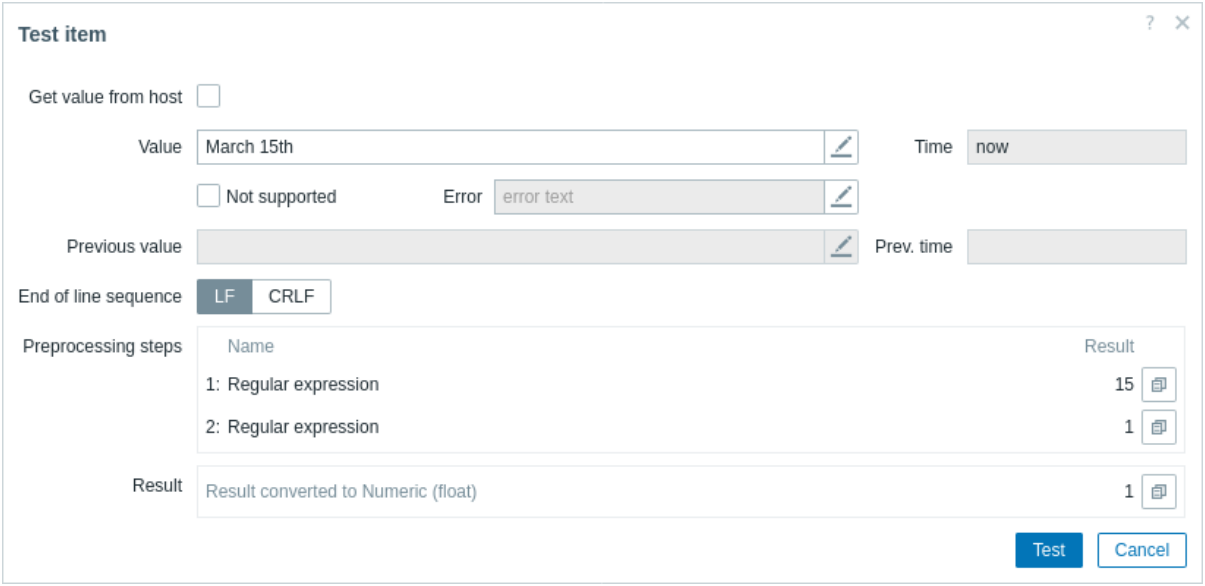


Możliwe jest testowanie:

- względem hipotetycznej wartości
- względem rzeczywistej wartości z hosta

Każdy krok przetwarzania wstępnego można testować osobno, a także wszystkie kroki można testować razem. Po kliknięciu przycisku *Test* lub *Test all steps* odpowiednio w bloku Actions zostanie otwarte okno testowania.

Testowanie hipotetycznej wartości



Parameter	Description
Pobierz wartość z host	Jeśli chcesz przetestować hipotetyczną wartość, pozostaw to pole wyboru niezaznaczone. Zobacz też: Testowanie rzeczywistej wartości.
Wartość	Wprowadź wartość wejściową do testu. Kliknięcie w pole parametru lub przycisk podglądu/edycji spowoduje otwarcie okna obszaru tekstowego do wprowadzania wartości lub bloku kodu.
Nieobsługiwane	Zaznacz to pole wyboru, aby przetestować nieobsługiwaną wartość. Ta opcja jest przydatna do testowania kroku przetwarzania wstępnego *Sprawdź wartość nieobsługiwaną*.

Parameter	Description
Błąd	Wprowadź tekst błędu. To pole jest aktywne, gdy *Pobierz wartość z host* jest odznaczone, ale *Nieobsługiwane* jest zaznaczone. Jeśli *Pobierz wartość z host* jest zaznaczone, to pole zostanie wypełnione rzeczywistym komunikatem błędu (tylko do odczytu) z host.
Czas	Wyświetlany jest czas wartości wejściowej: `now` (tylko do odczytu).
Poprzednia wartość	Wprowadź poprzednią wartość wejściową do porównania. Tylko dla kroków przetwarzania wstępnego *Change* i *Throttling*.
Poprzedni czas	Wprowadź czas poprzedniej wartości wejściowej do porównania. Tylko dla kroków przetwarzania wstępnego *Change* i *Throttling*. Wartość domyślna jest oparta na polu 'Update interval' pozycji (jeśli '1m', to to pole zostanie wypełnione `now-1m`). Jeśli nic nie zostanie określone lub użytkownik nie ma dostępu do host, domyślna wartość to `now-30s`.
Makra	Jeśli używane są jakiekolwiek makra, są one wyświetlane wraz z ich wartościami. Wartości można edytować na potrzeby testu, ale zmiany zostaną zapisane tylko w kontekście testowania.
Sekwencja końca linii	Wybierz sekwencję końca linii dla wielowierszowych wartości wejściowych: **LF** - sekwencja LF (line feed) **CRLF** - sekwencja CRLF (carriage-return line-feed).
Kroki przetwarzania wstępnego	Wyświetlane są kroki przetwarzania wstępnego; wynik testu jest pokazywany dla każdego kroku po kliknięciu przycisku *Test*. Wyniki testu są obcinane do maksymalnego rozmiaru 512KB podczas wysyłania do frontend. Wyniki testu można kopiować (nie więcej niż obcięte 512KB). Jeśli wynik został obcięty, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu myszą. Należy pamiętać, że dane większe niż 512KB są nadal w pełni przetwarzane przez serwer Zabbix. Jeśli krok zakończył się niepowodzeniem podczas testu, wyświetlana jest ikona błędu. Opis błędu jest wyświetlany po najechaniu myszą. W przypadku gdy dla kroku określono "Custom on fail" i ta akcja zostanie wykonana, bezpośrednio po wierszu kroku testu przetwarzania wstępnego pojawia się nowy wiersz pokazujący, jaka akcja została wykonana i jaki wynik wygenerowała (błąd lub wartość).
Wynik	Końcowy wynik testowania kroków przetwarzania wstępnego jest wyświetlany we wszystkich przypadkach, gdy wszystkie kroki są testowane razem (po kliknięciu przycisku *Test all steps*). Wyświetlany jest również typ konwersji do typu wartości pozycji, na przykład `Result converted to Numeric (unsigned)`. Wyniki testu są obcinane do maksymalnego rozmiaru 512KB podczas wysyłania do frontend. Końcowy wynik można kopiować (nie więcej niż obcięte 512KB). Jeśli wynik został obcięty, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu myszą. Należy pamiętać, że dane większe niż 512KB są nadal w pełni przetwarzane przez serwer Zabbix.

Kliknij *Test*, aby zobaczyć wynik po każdym kroku przetwarzania wstępnego.

Wartości testowe są przechowywane między sesjami testowymi dla poszczególnych kroków lub wszystkich kroków, co pozwala użytkownikowi zmieniać kroki przetwarzania wstępnego lub konfigurację pozycji, a następnie wrócić do okna testowania bez konieczności ponownego wprowadzania informacji. Wartości są jednak tracone po odświeżeniu strony.

Testowanie jest wykonywane przez serwer Zabbix. frontend wysyła odpowiednie żądanie do serwera i czeka na wynik. Żądanie zawiera wartość wejściową i kroki przetwarzania wstępnego (z rozwiniętymi makrami użytkownika). Dla kroków *Change* i *Throttling* można określić opcjonalną poprzednią wartość i czas. Serwer zwraca wyniki dla każdego kroku przetwarzania wstępnego.

Wszystkie błędy techniczne lub błędy walidacji danych wejściowych są wyświetlane w polu błędu u góry okna testowania.

Testowanie rzeczywistej wartości

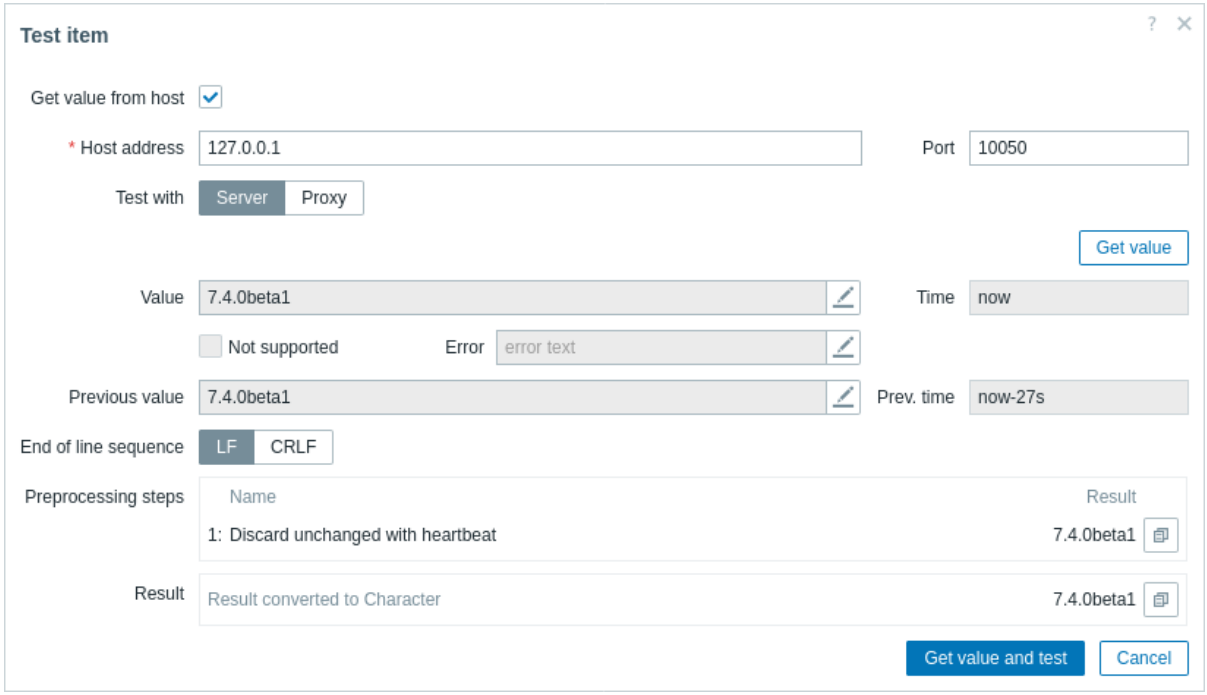
Aby przetestować preprocessing na rzeczywistej wartości:

- Zaznacz pole wyboru *Pobierz wartość z hosta*
- Wprowadź lub sprawdź parametry hosta (adres hosta, port, nazwę proxy/brak proxy) oraz szczegóły specyficzne dla pozycji (takie jak community SNMPv2 lub dane uwierzytelniające SNMPv3). Te pola uwzględniają kontekst:
  - Wartości są wstępnie wypełniane, gdy jest to możliwe, tj. w przypadku pozycji wymagających agenta informacje są pobierane z wybranego interfejsu agenta hosta
  - Wartości pozycji szablonu należy wprowadzić ręcznie
  - Wartości makr w postaci zwykłego tekstu są rozwiązywane
  - Jeśli wartość pola (lub jej część) jest sekretem lub makrem Vault, pole będzie puste i należy je wypełnić ręcznie. Jeśli dowolny parametr pozycji zawiera wartość makra sekretu, wyświetlany jest następujący komunikat ostrze-

gawczy: „Pozycja zawiera makra zdefiniowane przez użytkownika z wartościami sekretów. Wartości tych makr należy wprowadzić ręcznie.”

- Pola są wyłączone, gdy nie są potrzebne w kontekście typu pozycji (np. pola adresu hosta i proxy są wyłączone dla pozycji obliczanych)

- Kliknij *Pobierz wartość i przetestuj*, aby przetestować preprocessing



Jeśli w formularzu konfiguracji pozycji określono mapowanie wartości (pole *Pokaż wartość*), okno dialogowe testu pozycji wyświetli po wyniku końcowym dodatkowy wiersz o nazwie *Wynik z zastosowaną mapą wartości*.

Parametry specyficzne dla pobierania rzeczywistej wartości z hosta:

Parametr	Opis
Pobierz wartość z hosta	Zaznacz to pole wyboru, aby pobrać rzeczywistą wartość z hosta.
Adres hosta	Wprowadź adres hosta. To pole jest automatycznie wypełniane adresem interfejsu hosta pozycji.
Port	Wprowadź port hosta. To pole jest automatycznie wypełniane portem interfejsu hosta pozycji.
Dodatkowe pola dla interfejsów SNMP (wersja SNMP, community SNMP, nazwa kontekstu itp.)	Więcej informacji na temat konfigurowania interfejsu SNMP (v1, v2 i v3) zawiera sekcja Konfigurowanie monitorowania SNMP. Te pola są automatycznie wypełniane na podstawie interfejsu hosta pozycji.
Proxy	Określ proxy, jeśli host jest monitorowany przez proxy. To pole jest automatycznie wypełniane proxy hosta (jeśli istnieje).
Wartość	Wartość pobrana z hosta. Kliknięcie pola parametru lub przycisku wyświetlania/edycji otworzy okno obszaru tekstowego z wartością lub blokiem kodu. Wartości są skracane do maksymalnego rozmiaru 512 KB wyłącznie w frontendzie. Jeśli wynik zostanie skrócony, wyświetlana jest ikona ostrzeżenia. Opis ostrzeżenia jest wyświetlany po najechaniu kursorem myszy. Należy pamiętać, że dane większe niż 512 KB nadal są w całości przetwarzane przez serwer Zabbix.

Pozostałe parametry opisano powyżej w sekcji Testowanie hipotetycznej wartości.

Szczegóły przetwarzania wstępnego

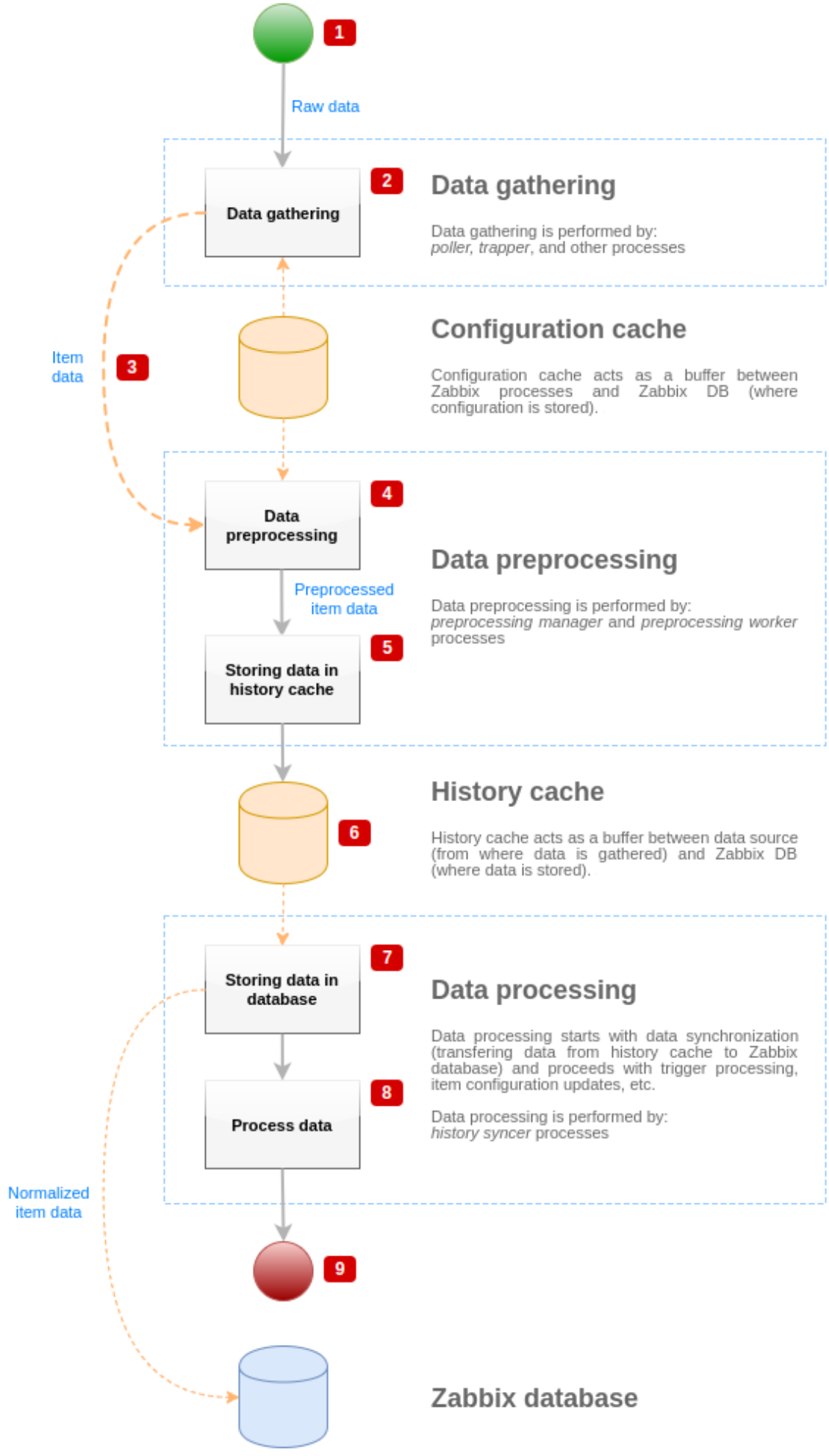
Przegląd

Ta sekcja zawiera szczegółowe informacje dotyczące wstępnego przetwarzania wartości pozycji. Wstępne przetwarzanie wartości pozycji umożliwia definiowanie i wykonywanie reguł transformacji dla odebranych wartości pozycji.

Wstępne przetwarzanie jest zarządzane przez proces menedżera wstępnego przetwarzania oraz procesy robocze, które wykonują poszczególne kroki wstępnego przetwarzania. Wszystkie wartości podlegające wstępnemu przetwarzaniu, odebrane z różnych modułów zbierających dane, przechodzą przez menedżera wstępnego przetwarzania przed dodaniem do pamięci podręcznej historii. Do komunikacji IPC między modułami zbierającymi dane (pollerami, trapperami itp.) a procesem wstępnego przetwarzania używana jest komunikacja oparta na gniazdach. Kroki wstępnego przetwarzania wykonuje serwer Zabbix lub proxy Zabbix (w przypadku pozycji monitorowanych przez proxy).

Przetwarzanie wartości pozycji

Aby zwizualizować przepływ danych od źródła danych do bazy danych Zabbix, możemy użyć następującego uproszczonego diagramu:


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

Powyższy diagram pokazuje tylko procesy, obiekty i działania związane z przetwarzaniem wartości pozycji w **uproszczonej** formie. Diagram nie pokazuje warunkowych zmian kierunku, obsługi błędów ani pętli. Nie pokazano również lokalnej pamięci podręcznej danych menedżera preprocessing, ponieważ nie wpływa ona bezpośrednio na przepływ danych. Celem tego diagramu jest pokazanie procesów biorących udział w przetwarzaniu wartości pozycji oraz sposobu ich współdziałania.

- Zbieranie danych rozpoczyna się od surowych danych ze źródła danych. Na tym etapie dane zawierają tylko identyfikator, znacznik czasu i wartość (może to być również wiele wartości).
- Niezależnie od tego, jaki typ mechanizmu zbierania danych jest używany, idea jest taka sama dla sprawdzeń aktywnych i pasywnych, dla pozycji trapper itd., ponieważ zmienia się tylko format danych i inicjator komunikacji (albo mechanizm zbierania danych oczekuje na połączenie i dane, albo inicjuje komunikację i żąda danych). Surowe dane są walidowane, konfiguracja pozycji jest pobierana z pamięci podręcznej konfiguracji (dane są wzbogacane o dane konfiguracyjne).
- Mechanizm IPC oparty na gniazdach jest używany do przekazywania danych z mechanizmów zbierania danych do menedżera preprocessing. W tym momencie mechanizm zbierania danych kontynuuje zbieranie danych bez oczekiwania na odpowiedź od menedżera preprocessing.
- Wykonywane jest preprocessing danych. Obejmuje to wykonanie kroków preprocessing oraz przetwarzanie pozycji zależnych.

Note:
Pozycja może zmienić swój stan na NOT SUPPORTED podczas wykonywania preprocessing, jeśli którykolwiek z kroków preprocessing zakończy się niepowodzeniem.

- Dane historyczne z lokalnej pamięci podręcznej danych menedżera preprocessing są opróżniane do pamięci podręcznej historii.
- W tym momencie przepływ danych zatrzymuje się do czasu następnej synchronizacji pamięci podręcznej historii (gdy proces synchronizacji historii wykonuje synchronizację danych).
- Proces synchronizacji rozpoczyna się od normalizacji danych przed zapisaniem ich w bazie danych Zabbix. Normalizacja danych wykonuje konwersje do żądanego typu pozycji (typu zdefiniowanego w konfiguracji pozycji), w tym obcinanie danych tekstowych na podstawie predefiniowanych rozmiarów dozwolonych dla tych typów (HISTORY_STR_VALUE_LEN dla string, HISTORY_TEXT_VALUE_LEN dla text oraz HISTORY_LOG_VALUE_LEN dla wartości log). Po zakończeniu normalizacji dane są wysyłane do bazy danych Zabbix.

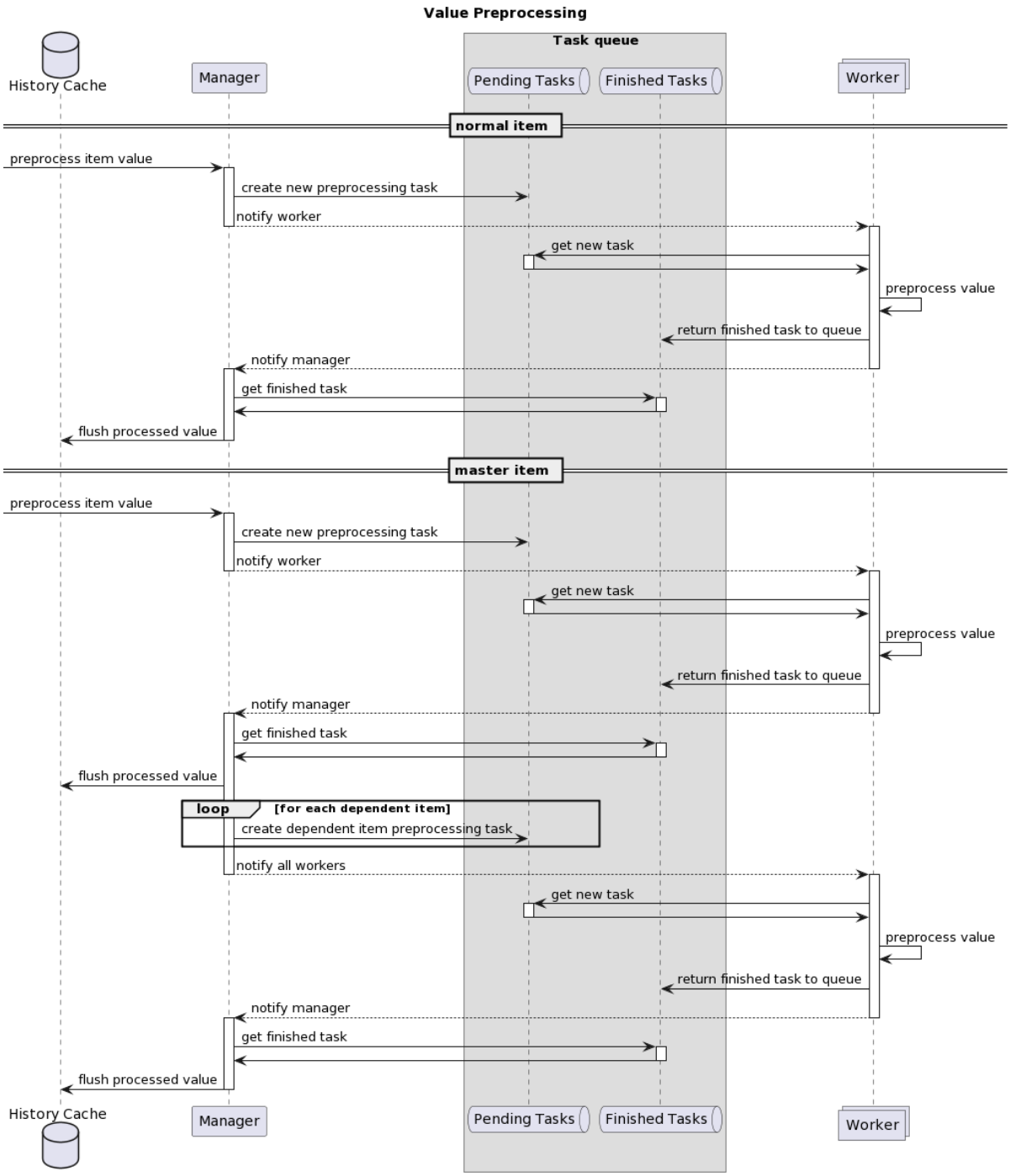
Note:
Pozycja może zmienić swój stan na NOT SUPPORTED, jeśli normalizacja danych zakończy się niepowodzeniem (na przykład gdy wartości tekstowej nie można przekonwertować na liczbę).

- Zebrane dane są przetwarzane — sprawdzane są wyzwalacze, konfiguracja pozycji jest aktualizowana, jeśli pozycja staje się NOT SUPPORTED itd.
- Jest to uznawane za koniec przepływu danych z punktu widzenia przetwarzania wartości pozycji.

Wstępne przetwarzanie wartości pozycji

Wstępne przetwarzanie danych odbywa się w następujących krokach:

- Jeśli pozycja nie ma ani wstępnego przetwarzania, ani pozycji zależnych, jej wartość jest albo dodawana do pamięci podręcznej historii, albo wysyłana do menedżera LLD. W przeciwnym razie wartość pozycji jest przekazywana do menedżera wstępnego przetwarzania za pomocą mechanizmu IPC opartego na gniazdach UNIX.
- Tworzone jest zadanie wstępnego przetwarzania, dodawane do kolejki, a workery wstępnego przetwarzania są powiadamiane o nowym zadaniu.
- W tym momencie przepływ danych zatrzymuje się, dopóki nie będzie dostępny co najmniej jeden nieużywany (tj. niewykonujący żadnych zadań) worker wstępnego przetwarzania.
- Gdy worker wstępnego przetwarzania jest dostępny, pobiera następne zadanie z kolejki.
- Po zakończeniu wstępnego przetwarzania (zarówno w przypadku nieudanego, jak i udanego wykonania kroków wstępnego przetwarzania) wstępnie przetworzona wartość jest dodawana do kolejki zakończonych zadań, a menedżer jest powiadamiany o nowym zakończonym zadaniu.
- Menedżer wstępnego przetwarzania konwertuje wynik do żądanego formatu (zdefiniowanego przez typ wartości pozycji) i albo dodaje go do pamięci podręcznej historii, albo wysyła do menedżera LLD.
- Jeśli dla przetwarzanej pozycji istnieją pozycje zależne, są one dodawane do kolejki wstępnego przetwarzania z użyciem wstępnie przetworzonej wartości nadrzędnej pozycji. Pozycje zależne są umieszczane w kolejce z pominięciem zwykłych żądań wstępnego przetwarzania wartości, ale tylko dla pozycji nadrzędnych z ustawioną wartością i niebędących w stanie NOT SUPPORTED.



Należy pamiętać, że na diagramie wstępne przetwarzanie pozycji nadrzędnej jest nieco uproszczone przez pominięcie buforowania wstępnego przetwarzania.

Kolejka przetwarzania wstępnego

Kolejka przetwarzania wstępnego jest zorganizowana następująco:

- lista oczekujących zadań:
  - zadania utworzone bezpośrednio na podstawie żądań przetwarzania wstępnego wartości, w kolejności ich otrzymania
- lista zadań natychmiastowych (przetwarzanych przed zadaniami oczekującymi):
  - zadania testowe (tworzone w odpowiedzi na żądania testowania pozycja/przetwarzania wstępnego przez frontend)
  - zadania zależnych pozycji
  - zadania sekwencyjne (zadania, które muszą być wykonywane w ścisłej kolejności):
    * posiadające kroki przetwarzania wstępnego wykorzystujące ostatnią wartość:
      · zmiana
      · ograniczanie
      · JavaScript (buforowanie kodu bajtowego)
    * buforowanie przetwarzania wstępnego zależnych pozycji
- lista zakończonych zadań

Buforowanie przetwarzania wstępnego

Buforowanie przetwarzania wstępnego zostało wprowadzone w celu poprawy wydajności przetwarzania wstępnego dla wielu zależnych pozycji mających podobne kroki przetwarzania wstępnego (co jest częstym rezultatem LLD).

Buforowanie polega na przetworzeniu wstępnym jednej zależnej pozycji i ponownym wykorzystaniu części wewnętrznych danych przetwarzania wstępnego dla pozostałych zależnych pozycji. Pamięć podręczna przetwarzania wstępnego jest obsługiwana tylko dla pierwszego kroku przetwarzania wstępnego następujących typów:

- Wzorzec Prometheus (indeksuje dane wejściowe według metryk)
- JSONPath (parsuje dane do drzewa obiektów i indeksuje pierwsze wyrażenie `[?(@.path == "value")]`)

Procesy robocze przetwarzania wstępnego

Plik konfiguracyjny serwera Zabbix umożliwia użytkownikom ustawienie liczby wątków roboczych przetwarzania wstępnego. Parametr konfiguracyjny StartPreprocessors należy użyć do ustawienia liczby uruchomionych wcześniej instancji procesów roboczych przetwarzania wstępnego, która powinna co najmniej odpowiadać liczbie dostępnych rdzeni CPU.

Jeśli zadania przetwarzania wstępnego nie są ograniczone przez CPU i obejmują częste żądania sieciowe, zaleca się skonfigurowanie dodatkowych procesów roboczych. Optymalną liczbę procesów roboczych przetwarzania wstępnego można określić na podstawie wielu czynników, w tym liczby "preprocessable" pozycji (pozycji, które wymagają wykonania dowolnych kroków przetwarzania wstępnego), liczby procesów zbierania danych, średniej liczby kroków dla przetwarzania wstępnego pozycji itp. Niewystarczająca liczba procesów roboczych może prowadzić do wysokiego użycia pamięci. W celu rozwiązywania problemów z nadmiernym użyciem pamięci w instalacji Zabbix, zobacz Profiling excessive memory usage with tcmalloc.

Zakładając jednak, że nie występują ciężkie operacje przetwarzania wstępnego, takie jak analizowanie dużych fragmentów XML/JSON, liczba procesów roboczych przetwarzania wstępnego może odpowiadać łącznej liczbie procesów zbierających dane. W ten sposób zazwyczaj (z wyjątkiem przypadków, gdy dane od zbierającego przychodzą zbiorczo) co najmniej jeden proces roboczy przetwarzania wstępnego będzie wolny dla zebranych danych.

Warning:
Zbyt duża liczba procesów zbierania danych (pollers, unreachable pollers, ODBC pollers, HTTP pollers, Java pollers, pingers, trappers, proxypollers) wraz z menedżerem IPMI, trapperem SNMP i procesami roboczymi przetwarzania wstępnego może wyczerpać limit deskryptorów plików na proces dla menedżera przetwarzania wstępnego.

 Wyczerpanie limitu deskryptorów plików na proces spowoduje zatrzymanie serwera Zabbix, zwykle krótko po uruchomieniu, ale czasami dopiero po dłuższym czasie. Aby uniknąć takich problemów, przejrzyj plik konfiguracyjny serwera Zabbix, aby zoptymalizować liczbę jednoczesnych kontroli i procesów. Dodatkowo, w razie potrzeby upewnij się, że limit deskryptorów plików jest ustawiony wystarczająco wysoko, sprawdzając i dostosowując limity systemowe.

Potok przetwarzania wartości

Przetwarzanie wartości pozycji jest wykonywane w wielu krokach (lub fazach) przez wiele procesów. Może to powodować:

- Zależna pozycja może otrzymywać wartości, podczas gdy wartość główna NIE może. Można to osiągnąć przy użyciu następującego przypadku użycia:
  - Główna pozycja ma typ wartości `UINT` (można użyć trapper item), zależna pozycja ma typ wartości `TEXT`.
  - Dla głównej i zależnej pozycji nie są wymagane żadne kroki preprocessing.
  - Wartość tekstowa (na przykład „abc") powinna zostać przekazana do głównej pozycji.
  - Ponieważ nie ma kroków preprocessing do wykonania, menedżer preprocessing sprawdza, czy główna pozycja nie jest w stanie NOT SUPPORTED oraz czy wartość jest ustawiona (oba warunki są spełnione), a następnie umieszcza zależną pozycję w kolejce z tą samą wartością co główna pozycja (ponieważ nie ma kroków preprocessing).
  - Gdy zarówno główna, jak i zależna pozycja osiągną fazę synchronizacji historii, główna pozycja przechodzi w stan NOT SUPPORTED z powodu błędu konwersji wartości (dane tekstowe nie mogą zostać przekonwertowane na liczbę całkowitą bez znaku).

W rezultacie zależna pozycja otrzymuje wartość, podczas gdy główna pozycja zmienia swój stan na NOT SUPPORTED.

- Zależna pozycja otrzymuje wartość, która nie jest obecna w historii głównej pozycji. Ten przypadek użycia jest bardzo podobny do poprzedniego, z wyjątkiem typu głównej pozycji. Na przykład, jeśli dla głównej pozycji używany jest typ `CHAR`, to wartość głównej pozycji zostanie obcięta w fazie synchronizacji historii, podczas gdy zależne pozycje otrzymają swoje wartości z początkowej (nieobciętej) wartości głównej pozycji.

Przykłady przetwarzania wstępnego

Przegląd

W tej sekcji przedstawiono przykłady użycia kroków przetwarzania wstępnego do realizacji niektórych praktycznych zadań.

Filtrowanie rekordów dziennika zdarzeń VMware

Ten przykład wykorzystuje krok przetwarzania wstępnego Matches regular expression do odfiltrowania niepotrzebnych zdarzeń z dziennika zdarzeń VMware.

1. Na działającym hoście VMware Hypervisor sprawdź, czy pozycja vmware.eventlog jest obecna i działa prawidłowo. Zwróć uwagę, że pozycja dziennika zdarzeń może już być obecna na hypervisorze, jeśli podczas tworzenia hosta został podłączony szablon VMware.

2. Na hoście VMware Hypervisor utwórz pozycję zależną typu *Log* i ustaw pozycję dziennika zdarzeń jako jej pozycję nadrzędną.

3. Na karcie *Preprocessing* pozycji zależnej kliknij *Add*, aby utworzyć krok przetwarzania wstępnego, i wybierz *Matches regular expression* z listy rozwijanej. Następnie określ jeden z następujących wzorców:

- Aby filtrować wszystkie zdarzenia dziennika:

```
.* logged in .*
```

- Aby filtrować wiersze zawierające nazwy użytkowników po "User":

```
\bUser\s+\K\S+
```

Attention:
Jeśli wyrażenie regularne nie zostanie dopasowane, pozycja zależna stanie się nieobsługiwana i zostanie wyświetlony odpowiedni komunikat o błędzie. Aby tego uniknąć, zaznacz pole wyboru *Custom on fail* i wybierz opcję, taką jak odrzucenie wartości lub ustawienie wartości niestandardowej. Pamiętaj, że wartości discarded nie są przechowywane w bazie danych; w rezultacie wyzwalacze nie są oceniane, a dane trendu nie są generowane.

Alternatywnie możesz użyć kroku przetwarzania wstępnego Regular expression, aby wyodrębnić pasujące grupy i kontrolować dane wyjściowe:

- Aby wyodrębnić i zwrócić całe zdarzenie dziennika zawierające "logged in", podaj następujące parametry:

```
Pattern: .*logged in.*
Output: \0
```

- Aby wyodrębnić i zwrócić nazwy użytkowników występujące po "User":

```
Pattern: User (.*?)(?=\ )
Output: \1
```

Sprawdzanie typu pobranej wartości

W tym przykładzie użyto kroku przetwarzania wstępnego Custom multiplier, aby sprawdzić, czy typ pobranej wartości pozycji jest numeryczny.

Na karcie *Preprocessing* pozycji wybierz krok przetwarzania wstępnego *Custom multiplier* i określ następujący parametr (mnoży pobraną wartość przez 1):

```
1
```

Attention:
Jeśli przetwarzanie wstępne zakończy się niepowodzeniem (np. dane wejściowe nie są numeryczne), pozycja stanie się nieobsługiwana i zostanie wyświetlony odpowiedni komunikat o błędzie. Aby tego uniknąć, zaznacz pole wyboru *Custom on fail* i wybierz opcję, taką jak odrzucenie wartości lub ustawienie wartości niestandardowej. Należy pamiętać, że wartości discarded nie są przechowywane w bazie danych; w rezultacie wyzwalacze nie są oceniane, a dane trendu nie są generowane.

Sprawdzanie nieobsługiwanej wartości

Ten przykład wykorzystuje krok preprocessing Check for not supported value do sprawdzenia, czy nie udało się pobrać wartości pozycji.

Gdy proces pollera serwera/proxy Zabbix próbuje pobrać wartość pozycji, może:

- Zwrócić prawidłowy wynik.
- Zwrócić wynik, który początkowo wydaje się prawidłowy, ale później może stać się nieobsługiwany (np. z powodu niezgodności typu wartości po preprocessing).
- Zwrócić błąd podczas pobierania wartości, powodując, że pozycja stanie się nieobsługiwana. Typowe przyczyny obejmują:
  - Nieznany klucz pozycji (dla pozycji Zabbix agent, Simple check lub Zabbix internal)
  - Nieznany OID (SNMP agent), nieznany sensor (IPMI agent) lub brak metryki JMX (JMX agent)
  - Nie można odczytać pliku trap (SNMP trap)
  - Nie znaleziono skryptu (External check)

- Taki URL nie istnieje (HTTP agent, Browser)
- Logowanie nie powiodło się (SSH agent, TELNET agent)
- Nieprawidłowa składnia formuły (Calculated), błąd składni JavaScript (Script) lub nieprawidłowe SQL (Database monitor)

Aby wykrywać i obsługiwać błędy pobierania wartości pozycji, można użyć kroku preprocessing *Check for not supported value*. Należy pamiętać, że ten krok jest zawsze wykonywany jako pierwszy i wykrywa tylko błędy, które występują przed rozpoczęciem preprocessing.

Na karcie *Preprocessing* pozycji wybierz krok preprocessing *Check for not supported value* i określ jeden z następujących parametrów:

- Dla dowolnych błędów:

```
Parameter: any error
```

- Dla błędów zawierających "cannot connect":

```
Parameter: error matches
Pattern: (?i)cannot connect
```

Następnie użyj opcji *Custom on fail*, aby odrzucić wartość (w tym przypadku błąd), ustawić wartość niestandardową lub zwrócić niestandardowy komunikat o błędzie. Pamiętaj, że wartości discarded nie są przechowywane w bazie danych; w rezultacie wyzwalacze nie są obliczane, a dane trendu nie są generowane.

Funkcjonalność JSONPath

Przegląd

Ta sekcja przedstawia obsługiwaną funkcjonalność JSONPath w krokach wstępnego przetwarzania wartości pozycji.

JSONPath składa się z segmentów oddzielonych kropkami. Segment może mieć postać prostego słowa, reprezentującego nazwę wartości JSON, znaku wieloznacznego (`*`) lub bardziej złożonej konstrukcji ujętej w nawiasy kwadratowe. Kropka przed segmentem ujętym w nawiasy jest opcjonalna i może zostać pominięta.

Przykład JSONPath	Opis
`$.object.name`	Zwraca zawartość `object.name`.
`$.object['name']`	Zwraca zawartość `object.name`.
`$.object.['name']`	Zwraca zawartość `object.name`.
`$["object"]['name']`	Zwraca zawartość `object.name`.
`$.['object'].["name"]`	Zwraca zawartość `object.name`.
`$.object.history.length()`	Zwraca liczbę elementów tablicy `object.history`.
`$[?(@.name == 'Object')].price.first()`	Zwraca wartość właściwości `price` z pierwszego obiektu o nazwie „Object".
`$[?(@.name == 'Object')].history.first().length()`	Zwraca liczbę elementów tablicy historii z pierwszego obiektu o nazwie „Object".
`$[?(@.price > 10)].length()`	Zwraca liczbę obiektów z ceną większą niż 10.

Zobacz także: Escaping special characters from LLD macro values in JSONPath.

Obsługiwane segmenty

Segment	Opis
`<name>`	Dopasowuje właściwość obiektu według nazwy.
`*`	Dopasowuje wszystkie właściwości obiektu.
`['<name>']`	Dopasowuje właściwość obiektu według nazwy.
`['<name>', '<name>', ...]`	Dopasowuje właściwość obiektu według dowolnej z podanych nazw.
`[<index>]`	Dopasowuje element tablicy według indeksu.
`[<number>, <number>, ...]`	Dopasowuje element tablicy według dowolnego z podanych indeksów.
`[*]`	Dopasowuje wszystkie właściwości obiektu lub elementy tablicy.

Segment	Opis
`[<start>:<end>]`	Dopasowuje elementy tablicy według zdefiniowanego zakresu: **<start>** - pierwszy dopasowywany indeks (włącznie); jeśli nie został określony, dopasowywane są wszystkie elementy tablicy od początku; jeśli jest ujemny, określa przesunięcie początkowe liczone od końca tablicy; **<end>** - ostatni dopasowywany indeks (z wyłączeniem); jeśli nie został określony, dopasowywane są wszystkie elementy tablicy do końca; jeśli jest ujemny, określa przesunięcie początkowe liczone od końca tablicy.
`[?(<expression>)]`	Dopasowuje obiekty/elementy tablicy przez zastosowanie wyrażenia filtra.

Aby znaleźć pasujący segment z pominięciem jego pochodzenia (segment odłączony), należy poprzedzić go dwiema kropkami (`..`). Na przykład `$..name` lub `$..['name']` zwracają wartości wszystkich właściwości `name`.

Nazwy dopasowanych elementów można wyodrębnić, dodając do JSONPath sufiks tyldy (~). Zwraca on nazwę dopasowanego obiektu lub indeks dopasowanego elementu tablicy w formacie ciągu znaków. Format wyjściowy jest zgodny z tymi samymi zasadami co w przypadku innych zapytań JSONPath — wyniki ścieżki określonej są zwracane „tak jak są", a wyniki ścieżki nieokreślonej są zwracane w tablicy. Jednak wyodrębnianie nazwy elementu dopasowanego przez ścieżkę określoną ma niewielką wartość, ponieważ jest ona już znana.

Wyrażenie filtra

Wyrażenie filtra jest wyrażeniem arytmetycznym w notacji infiksowej.

Obsługiwane operandy:

Operand	Opis

`"<text>"`
`'<text>'`
	Stała tekstowa.
	
	Przykład:
	`'value: \'1\''`
	`"value: '1'"`

`<number>`	Stała numeryczna obsługująca notację naukową. Przykład: 123
`<jsonpath starting with $>`	Wartość wskazana przez JSONPath od węzła głównego dokumentu wejściowego; obsługiwane są tylko ścieżki jednoznaczne. Przykład: `$.object.name`
`<jsonpath starting with @>`	Wartość wskazana przez JSONPath od bieżącego obiektu/elementu; obsługiwane są tylko ścieżki jednoznaczne. Przykład: `@.name`

Obsługiwane operatory: |Operator|Typ|Opis|Wynik| |-|---|---|---| |−|Binary|Odejmowanie|Liczba| |+|Binary|Dodawanie|Liczba| |/|Binary|Dzielenie|Liczba| |*|Binary|Mnożenie|Liczba| |==|Binary|Równość|Boolean (1/0)| |!=|Binary|Nierówność|Boolean (1/0)| |<|Binary|Mniejsze niż|Boolean (1/0)| |<=|Binary|Mniejsze lub równe|Boolean (1/0)| |>|Binary|Większe niż|Boolean (1/0)| |>=|Binary|Większe lub równe|Boolean (1/0)| |=~|Binary|Dopasowanie do wyrażenia regularnego|Boolean (1/0)| |!|Unary|Logiczne NIE|Boolean (1/0)| ||||Binary|Logiczne LUB|Boolean (1/0)| |&&|Binary|Logiczne I|Boolean (1/0)|

Funkcje

Funkcje mogą być używane na końcu JSONPath. Można łączyć wiele funkcji w łańcuch, jeśli poprzedzająca funkcja zwraca wartość akceptowaną przez następną funkcję.

Obsługiwane funkcje:

Funkcja	Opis	Wejście	Wyjście
`avg`	Średnia wartość liczb w tablicy wejściowej	Tablica liczb	Liczba
`min`	Minimalna wartość liczb w tablicy wejściowej	Tablica liczb	Liczba
`max`	Maksymalna wartość liczb w tablicy wejściowej	Tablica liczb	Liczba
`sum`	Suma liczb w tablicy wejściowej	Tablica liczb	Liczba
`length`	Liczba elementów w tablicy wejściowej	Tablica	Liczba

Funkcja	Opis	Wejście	Wyjście
`first`	Pierwszy element tablicy	Tablica	Konstrukcja JSON (obiekt, tablica, wartość) zależnie od zawartości tablicy wejściowej

Funkcje agregujące JSONPath akceptują ujęte w cudzysłów wartości liczbowe. Wartości te są automatycznie konwertowane z ciągów znaków na typy liczbowe, gdy wymagana jest agregacja. Niezgodne dane wejściowe spowodują wygenerowanie błędu przez funkcję.

Wartość wyjściowa

Ścieżki JSONPath można podzielić na ścieżki określone i nieokreślone. Ścieżka określona może zwrócić tylko null lub jedno dopasowanie. Ścieżka nieokreślona może zwrócić wiele dopasowań: JSONPath z odłączonymi segmentami, wieloma listami nazw/indeksów, wycinkami tablic lub segmentami wyrażeń. Jednak gdy używana jest funkcja, JSONPath staje się ścieżką określoną, ponieważ funkcje zawsze zwracają pojedynczą wartość.

Ścieżka określona zwraca obiekt/tablicę/wartość, do których się odwołuje. Natomiast ścieżka nieokreślona zwraca tablicę dopasowanych obiektów/tablic/wartości.

> **Attention:**
> Kolejność właściwości w wynikach zapytania JSONPath może nie odpowiadać oryginalnej kolejności właściwości w JSON ze względu na wewnętrzne metody optymalizacji. Na przykład JSONPath `$.books[1]["author", "title"]` może zwrócić `["title", "author"]`. Jeśli zachowanie oryginalnej kolejności właściwości jest istotne, należy rozważyć alternatywne metody przetwarzania po wykonaniu zapytania.

Zasady formatowania ścieżki

Białe znaki (spacja, znak tabulacji) mogą być używane w segmentach notacji nawiasowej i wyrażeniach, na przykład: `$[ 'a' ][ 0 ][ ?( $.b == 'c' ) ][ : -1 ].first( )`.

Ciągi znaków powinny być ujmowane w pojedyncze (') lub podwójne (") cudzysłowy. Wewnątrz ciągów znaków pojedyncze lub podwójne cudzysłowy (w zależności od tego, które są używane do ich ujmowania) oraz ukośniki odwrotne (\) są poprzedzane znakiem ukośnika odwrotnego (\).

Przykład

```
{
  "books": [
    {
      "category": "reference",
      "author": "Nigel Rees",
      "title": "Powiedzenia stulecia",
      "price": 8.95,
      "id": 1
    },
    {
      "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Miecz honoru",
      "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": "Władca Pierścieni",
      "isbn": "0-395-19395-8",
```

```
      "price": 22.99,
      "id": 4
    }
  ],
  "services": {
    "delivery": {
      "servicegroup": 1000,
      "description": "Dostawa następnego dnia w lokalnym mieście",
      "active": true,
      "price": 5
    },
    "bookbinding": {
      "servicegroup": 1001,
      "description": "Drukowanie i oprawa książki w formacie A5",
      "active": true,
      "price": 154.99
    },
    "restoration": {
      "servicegroup": 1002,
      "description": "Różne metody renowacji",
      "active": false,
      "methods": [
        {
          "description": "Czyszczenie chemiczne",
          "price": 46
        },
        {
          "description": "Prasowanie stron uszkodzonych przez wilgoć",
          "price": 24.5
        },
        {
          "description": "Ponowne oprawienie rozdartej książki",
          "price": 99.49
        }
      ]
    }
  },
  "filters": {
    "price": 10,
    "category": "fiction",
    "no filters": "no \"filters\""
  },
  "closed message": "Sklep jest zamknięty",
  "tags": [
    "a",
    "b",
    "c",
    "d",
    "e"
  ]
}
```


JSONPath	Type	Result
`$.filters.price`	definite	10
`$.filters.category`	definite	fiction
`$.filters['no filters']`	definite	no ”filters”

`$.filters`	definite	{
		”price”: 10,
		”category”: ”fiction”,
		”no filters”: ”no \”filters\””
		}

`$.books[1].title`	definite	Miecz honoru

JSONPath	Type	Result
`$.books[-1].author`	definite	J. R. R. Tolkien
`$.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”, ”Powiedzenia stulecia”]
`$.books[1]['author', "title"]`	indefinite	[”Miecz honoru”, ”Evelyn Waugh”]
`$..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	[”Miecz honoru”] Uwaga: To zapytanie pokazuje, że w zapytaniach można używać działań arytmetycznych; można je uprościć do `$.books[?(@.id == 2)].title`
`$.books[?(@.id == 2 || @.id == 4)].title`	indefinite	[”Miecz honoru”, ”Władca Pierścieni”]
`$.books[?(!(@.id == 2))].title`	indefinite	[”Powiedzenia stulecia”, ”Moby Dick”, ”Władca Pierścieni”]
`$.books[?(@.id != 2)].title`	indefinite	[”Powiedzenia stulecia”, ”Moby Dick”, ”Władca Pierścieni”]
`$.books[?(@.title =~ " of ")].title`	indefinite	[”Powiedzenia stulecia”, ”Miecz honoru”, ”Władca Pierścieni”]
`$.books[?(@.price > 12.99)].title`	indefinite	[”Władca Pierścieni”]
`$.books[?(@.author > "Herman Melville")].title`	indefinite	[”Powiedzenia stulecia”, ”Władca Pierścieni”]
`$.books[?(@.price > $.filters.price)].title`	indefinite	[”Miecz honoru”, ”Władca Pierścieni”]
`$.books[?(@.category == $.filters.category)].title`	indefinite	[”Miecz honoru”,”Moby Dick”,”Władca Pierścieni”]
`$.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”: ”Powiedzenia stulecia”
		},
		{
		”price”: 12.99,
		”id”: 2,
		”category”: ”fiction”,
		”author”: ”Evelyn Waugh”,
		”title”: ”Miecz honoru”
		},
		{
		”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”: ”Władca Pierścieni”,
		”isbn”: ”0-395-19395-8”
		}
		]

`$.services..[?(@.price > 50)].description`	indefinite	[”Drukowanie i oprawa książki w formacie A5”, ”Ponowne oprawienie rozdartej książki”]
`$..id.length()`	definite	4
`$.books[?(@.id == 2)].title.first()`	definite	Miecz honoru
`$..tags.first().length()`	definite	5 Uwaga: `$..tags` jest ścieżką niedefinitywną, więc zwraca tablicę dopasowanych elementów, tj. `[["a", "b", "c", "d", "e" ]]`; `first()` zwraca pierwszy element, tj. `["a", "b", "c", "d", "e"]`; `length()` oblicza długość elementu, tj. 5.
`$.books[*].price.min()`	definite	8.95
`$..price.max()`	definite	154.99
`$.books[?(@.category == "fiction")].price.avg()`	definite	14.99
`$.books[?(@.category == $.filters.xyz)].title`	indefinite	Uwaga: Zapytanie bez dopasowania zwraca NULL dla ścieżek definitywnych i niedefinitywnych.
`$.services[?(@.active=="true")].servicegroup`	indefinite	[1001,1000] Uwaga: W porównaniach wartości logicznych należy używać stałych tekstowych.
`$.services[?(@.active=="false")].servicegroup`	indefinite	[1002] Uwaga: W porównaniach wartości logicznych należy używać stałych tekstowych.
`$.services[?(@.servicegroup=="1002")].description`	definite	test

Escapowanie znaków specjalnych z wartości makr LLD w JSONPath

Gdy makra wykrywania niskiego poziomu są używane w przetwarzaniu JSONPath i ich wartości są rozwijane, stosowane są następujące zasady escapowania znaków specjalnych:

- tylko znak ukośnika odwrotnego (\) i cudzysłowu (”) są brane pod uwagę przy escapowaniu;
- jeśli rozwinięta wartość makra zawiera te znaki, każdy z nich jest poprzedzany ukośnikiem odwrotnym;
- jeśli są już poprzedzone ukośnikiem odwrotnym, nie jest to traktowane jako escapowanie i zarówno ukośnik odwrotny, jak i następujące po nim znaki specjalne są escapowane ponownie.

Na przykład:

JSONPath	LLD macro value	After substitution
$.[?(@.value == ”{#MACRO}”)]	special ”value”	$.[?(@.value == ”special \”value\””)]
	c:\temp	$.[?(@.value == ”c:\\temp”)]
	a\\b	$.[?(@.value == ”a\\\\b”)]

Gdy makro jest używane w wyrażeniu, makro, które może zawierać znaki specjalne, powinno być ujęte w cudzysłów:

JSONPath	LLD macro value	After substitution	Result
$.[?(@.value == ”{#MACRO}”)]	special ”value”	$.[?(@.value == ”special \”value\””)]	OK
$.[?(@.value == {#MACRO})]		$.[?(@.value == special \”value\”)]	**Bad JSONPath expression**

Gdy makro jest używane w ścieżce, makro, które może zawierać znaki specjalne, powinno być ujęte w nawiasy kwadratowe **i** cudzysłów:

JSONPath	LLD macro value	After substitution	Result
$.[”{#MACRO}”].value	c:\temp	$.[”c:\\temp”].value	OK
$.{#MACRO}.value		$.c:\\temp.value	**Bad JSONPath expression**

Wstępne przetwarzanie JavaScript

Przegląd

Ta sekcja zawiera szczegółowe informacje dotyczące preprocessingu za pomocą JavaScript.

> **Attention:**
> Nie używaj niezadeklarowanych przypisań w JavaScript używanym do preprocessingu. Użyj `var`, aby deklarować zmienne lokalne.

Wstępne przetwarzanie JavaScript

Wstępne przetwarzanie JavaScript jest wykonywane przez wywołanie funkcji JavaScript z jednym parametrem ’value’ oraz treścią funkcji dostarczoną przez użytkownika. Wynik kroku wstępnego przetwarzania jest wartością zwracaną przez tę funkcję, na przykład aby wykonać konwersję z Fahrenheita na Celsjusza, wpisz:

```
return (value - 32)  * 5 / 9
```

w parametrach wstępnego przetwarzania JavaScript, które zostaną opakowane przez serwer w funkcję JavaScript:

```
function (value)
{
   return (value - 32) * 5 / 9
}
```

Parametr wejściowy ’value’ jest zawsze przekazywany jako ciąg znaków. Wartość zwracana jest automatycznie przekształcana do ciągu znaków za pomocą metody toString() (jeśli się to nie powiedzie, błąd jest zwracany jako wartość ciągu znaków), z kilkoma wyjątkami:

- zwrócenie wartości undefined spowoduje błąd;
- zwrócenie wartości null spowoduje odrzucenie wartości wejściowej, podobnie jak w przypadku wstępnego przetwarzania 'Discard value' dla akcji 'Custom on fail'.

Błędy mogą być zwracane przez zgłaszanie wartości/obiektów (zwykle ciągów znaków lub obiektów Error).

Na przykład:

```
if (value == 0)
    throw "Zero input value"
return 1/value
```

Każdy skrypt ma 10-sekundowy limit czasu wykonania (w zależności od skryptu wyzwolenie limitu może nastąpić później); jego przekroczenie spowoduje zwrócenie błędu. Wymuszany jest limit sterty wynoszący 512 megabajtów.

Kod bajtowy kroku wstępnego przetwarzania JavaScript jest buforowany i ponownie używany przy następnym zastosowaniu kroku. Wszelkie zmiany w krokach wstępnego przetwarzania pozycja spowodują zresetowanie buforowanego skryptu i jego późniejszą rekompilację.

Kolejne niepowodzenia w czasie wykonywania (3 z rzędu) spowodują ponowne zainicjowanie silnika, aby ograniczyć możliwość, że jeden skrypt zakłóci środowisko wykonawcze kolejnych skryptów (ta akcja jest rejestrowana przy DebugLevel 4 i wyższym).

Wstępne przetwarzanie JavaScript jest zaimplementowane przy użyciu silnika JavaScript Duktape.

> **Note:**
> Wbudowanych definicji metod obiektów JavaScript nie można redefiniować w skryptach wstępnego przetwarzania pozycja.

Zobacz też: Additional JavaScript objects and global functions

Używanie makr w skryptach

Możliwe jest używanie makr użytkownika (a także makr LLD w kontekście wykrywania niskiego poziomu) w kodzie JavaScript. Jeśli skrypt zawiera makra użytkownika, makra te są rozwiązywane przez serwer/proxy przed wykonaniem określonych kroków preprocessing. Należy pamiętać, że podczas testowania kroków preprocessing w frontend wartości makr nie będą pobierane i trzeba je wprowadzić ręcznie.

> **Note:**
> Kontekst jest ignorowany, gdy makro jest zastępowane swoją wartością. Wartość makra jest wstawiana do kodu w niezmienionej postaci; nie ma możliwości dodania dodatkowego escaping przed umieszczeniem wartości w kodzie JavaScript. Należy pamiętać, że w niektórych przypadkach może to powodować błędy JavaScript.

W poniższym przykładzie, jeśli odebrana wartość przekracza wartość makra {$THRESHOLD}, zamiast niej zostanie zwrócona wartość progu (jeśli jest obecna):

```
var threshold = '{$THRESHOLD}';
return (!isNaN(threshold) && value > threshold) ? threshold : value;
```

Przykłady

Poniższe przykłady ilustrują, jak można używać preprocessingu JavaScript.

Każdy przykład zawiera krótki opis, treść funkcji dla parametrów preprocessingu JavaScript oraz wynik kroku preprocessingu — wartość zwracaną przez funkcję.

Przykład 1: Konwersja liczby (notacja naukowa na liczbę całkowitą)

Przekonwertuj liczbę „2.62128e+07" z notacji naukowej na liczbę całkowitą.

```
return (Number(value))
```

Wartość zwrócona przez funkcję: 26212800.

Przykład 2: Konwersja liczby (z systemu binarnego na dziesiętny)

Przekonwertuj liczbę binarną „11010010" na liczbę dziesiętną.

```
return(parseInt(value,2))
```

Wartość zwrócona przez funkcję: 210.

Przykład 3: Zaokrąglanie liczby

Zaokrąglij liczbę „18.2345" do 2 cyfr.

```
return(Math.round(value* 100) / 100)
```

Wartość zwrócona przez funkcję: 18.23.

Przykład 4: Zliczanie liter w ciągu znaków

Policz liczbę liter w ciągu znaków „Zabbix”.

```
return (value.length)
```

Wartość zwrócona przez funkcję: 6.

Przykład 5: Pobierz pozostały czas

Pobierz pozostały czas (w sekundach) do daty wygaśnięcia certyfikatu (Feb 12 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);
```

Wartość zwracana przez funkcję: 44380233.

Przykład 6: Usuń właściwości JSON

Zmodyfikuj strukturę danych JSON, usuwając wszystkie właściwości z kluczem "data_size" lub "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)
```

Wartość przyjmowana przez funkcję:

```
[
    {
        "table_name":"history",
        "data_size":"326.05",
        "index_size":"174.34"
    },
    {
        "table_name":"history_log",
        "data_size":"6.02",
        "index_size":"3.45"
    }
]
```

Wartość zwracana przez funkcję:

```
[
    {
        "table_name":"history"
    },
    {
        "table_name":"history_log"
    }
]
```

Przykład 7: Konwersja statusu Apache do formatu JSON

Przekonwertuj wartość otrzymaną z pozycji Zabbix agent web.page.get (np. web.page.get[http://127.0.0.1:80/server-status?auto]) na obiekt JSON.

```
// Konwersja statusu Apache do formatu JSON

// Podziel wartość na podciągi i umieść te podciągi w tablicy
var lines = value.split('\n');

// Utwórz pusty obiekt "output"
var output = {};

// Utwórz obiekt "workers" ze wstępnie zdefiniowanymi właściwościami
var workers = {
    '_': 0, 'S': 0, 'R': 0, 'W': 0,
    'K': 0, 'D': 0, 'C': 0, 'L': 0,
    'G': 0, 'I': 0, '.': 0
};

// Dodaj podciągi z tablicy "lines" do obiektu "output" jako właściwości (pary klucz-wartość)
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]);
    }
}

// Metryki dla wielu wersji
output.ServerUptimeSeconds = output.ServerUptimeSeconds || output.Uptime;
output.ServerVersion = output.ServerVersion || output.Server;

// Przeanalizuj właściwość "Scoreboard", aby uzyskać liczbę workerów
if (typeof output.Scoreboard === 'string') {
    for (var i = 0; i < output.Scoreboard.length; i++) {
        var char = output.Scoreboard[i];

        workers[char]++;
    }
}

// Dodaj dane workerów do obiektu "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['.']
};

// Zwróć ciąg JSON
return JSON.stringify(output);
```

Wartość akceptowana przez funkcję:

```
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__.....................................................................................
```

Wartość zwracana przez funkcję:


```
{
    "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
    }
}
```

Dodatkowe obiekty JavaScript

Przegląd

Ta sekcja opisuje dodatki Zabbix do języka JavaScript zaimplementowanego przy użyciu Duktape oraz obsługiwane globalne funkcje JavaScript.

Attention:
Nie używaj niezadeklarowanych przypisań w skryptach JavaScript przetwarzania wstępnego. Używaj `var` do deklarowania zmiennych lokalnych.

Wbudowane obiekty

Zabbix

Obiekt Zabbix umożliwia interakcję z wewnętrzną funkcjonalnością Zabbix.

Method	Description
`log(loglevel, message)`	Zapisuje <message> do logu Zabbix z użyciem poziomu logowania <loglevel> (zobacz parametr DebugLevel w pliku konfiguracyjnym).

Przykład:

```
Zabbix.log(3, "this is a log entry written with 'Warning' log level")
```

Możesz używać następujących aliasów:

Alias	Alias to
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:
Łączny rozmiar wszystkich rejestrowanych komunikatów jest ograniczony do 8 MB na jedno wykonanie skryptu.

Method	Description
`sleep(delay)`	Opóźnia wykonanie JavaScript o `delay` milisekund.

Przykład (opóźnienie wykonania o 15 sekund):

```
Zabbix.sleep(15000)
```

HttpRequest

Ten obiekt hermetyzuje uchwyt cURL, umożliwiając wykonywanie prostych żądań HTTP. Błędy są zgłaszane jako wyjątki.

Attention:
Inicjalizacja wielu obiektów `HttpRequest` jest ograniczona do 10 na wykonanie skryptu.

Method	Description
`addHeader(value)`	Dodaje pole nagłówka HTTP. To pole jest używane dla wszystkich kolejnych żądań do momentu wyczyszczenia za pomocą metody `clearHeader()`. Łączna długość pól nagłówka, które można dodać do pojedynczego obiektu `HttpRequest`, jest ograniczona do 128 KB (wliczając znaki specjalne i nazwy nagłówków).
`clearHeader()`	Czyści nagłówek HTTP. Jeśli nie ustawiono żadnych pól nagłówka, `HttpRequest` ustawi Content-Type na application/json, jeśli wysyłane dane są sformatowane jako JSON; w przeciwnym razie na text/plain.
`connect(url)`	Wysyła żądanie HTTP CONNECT do adresu URL i zwraca odpowiedź.
`customRequest(method, url, data)`	Umożliwia określenie dowolnej metody HTTP w pierwszym parametrze. Wysyła żądanie określonej metody do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`delete(url, data)`	Wysyła żądanie HTTP DELETE do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`getHeaders(<asArray>)`	Zwraca obiekt odebranych pól nagłówka HTTP. Parametr `asArray` może mieć wartość "true" (np. `getHeaders(true)`), "false" lub być niezdefiniowany. Jeśli ma wartość "true", wartości odebranych pól nagłówka zostaną zwrócone jako tablice; należy użyć tej opcji w celu pobrania wartości wielu nagłówków o tej samej nazwie. Jeśli parametr nie jest ustawiony lub ma wartość "false", wartości odebranych pól nagłówka zostaną zwrócone jako ciągi znaków.
`get(url, data)`	Wysyła żądanie HTTP GET do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`head(url)`	Wysyła żądanie HTTP HEAD do adresu URL i zwraca odpowiedź.
`options(url)`	Wysyła żądanie HTTP OPTIONS do adresu URL i zwraca odpowiedź.
`patch(url, data)`	Wysyła żądanie HTTP PATCH do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`put(url, data)`	Wysyła żądanie HTTP PUT do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`post(url, data)`	Wysyła żądanie HTTP POST do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.
`getStatus()`	Zwraca kod stanu ostatniego żądania HTTP.
`setProxy(proxy)`	Ustawia proxy HTTP na wartość "proxy". Jeśli ten parametr jest pusty, proxy nie jest używane.

`setHttpAuth(bitmask, username, password)`	Ustawia w parametrze 'bitmask' włączone metody uwierzytelniania HTTP (HTTPAUTH_BASIC, HTTPAUTH_DIGEST, HTTPAUTH_NEGOTIATE, HTTPAUTH_NTLM, HTTPAUTH_NONE).
	Flaga HTTPAUTH_NONE umożliwia wyłączenie uwierzytelniania HTTP.
	Przykłady:
	`request.setHttpAuth(HTTPAUTH_NTLM | HTTPAUTH_BASIC, username, password)`
	`request.setHttpAuth(HTTPAUTH_NONE)`

`trace(url, data)`	Wysyła żądanie HTTP TRACE do adresu URL z opcjonalnym ładunkiem *data* i zwraca odpowiedź.

Przykład:

```
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

Obiekt XML umożliwia przetwarzanie danych XML we wstępnym przetwarzaniu pozycji i wykrywaniu niskopoziomowym oraz w webhookach.

Attention:
Aby używać obiektu XML, serwer/proxy musi być skompilowany z obsługą libxml2.

Method	Description
`XML.query(data, expression)`	Pobiera zawartość węzła przy użyciu XPath. Zwraca null, jeśli węzeł nie zostanie znaleziony. **expression** - wyrażenie XPath; **data** - dane XML jako ciąg znaków.
`XML.toJson(data)`	Konwertuje dane w formacie XML do JSON.
`XML.fromJson(object)`	Konwertuje dane w formacie JSON do XML.

Przykład:

Wejście:

```
<menu>
    <food type = "breakfast">
        <name>Chocolate</name>
        <price>$5.95</price>
        <description></description>
        <calories>650</calories>
    </food>
</menu>
```

Wyjście:

```
{
    "menu": {
        "food": {
            "@type": "breakfast",
            "name": "Chocolate",
            "price": "$5.95",
            "description": null,
            "calories": "650"
        }
    }
}
```

Zasady serializacji

Konwersja XML do JSON będzie przetwarzana zgodnie z następującymi zasadami (w przypadku konwersji JSON do XML stosowane są zasady odwrotne):

1. Atrybuty XML zostaną przekonwertowane na klucze, których nazwy są poprzedzone znakiem ’@’.

Przykład:

Wejście:

```
<xml foo="FOO">
  <bar>
    <baz>BAZ</baz>
  </bar>
</xml>
```

Wyjście:

```
{
  "xml": {
    "@foo": "FOO",
    "bar": {
      "baz": "BAZ"
    }
  }
}
```

2. Elementy samozamykające się (<foo/>) zostaną przekonwertowane tak, jakby miały wartość ’null’.

Przykład:

Wejście:

```
<xml>
  <foo/>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": null
  }
}
```

3. Puste atrybuty (o wartości ”“) zostaną przekonwertowane tak, jakby miały wartość pustego ciągu znaków (”).

Przykład:

Wejście:

```
<xml>
  <foo bar="" />
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": {
      "@bar": ""
    }
  }
}
```

4. Wiele węzłów podrzędnych o tej samej nazwie elementu zostanie przekonwertowanych na pojedynczy klucz, którego wartością będzie tablica wartości.

Przykład:

Wejście:

```
<xml>
  <foo>BAR</foo>
  <foo>BAZ</foo>
  <foo>QUX</foo>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": ["BAR", "BAZ", "QUX"]
  }
}
```

5. Jeśli element tekstowy nie ma atrybutów ani elementów podrzędnych, zostanie przekonwertowany jako ciąg znaków.

Przykład:

Wejście:

```
<xml>
    <foo>BAZ</foo>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": "BAZ"
   }
}
```

6. Jeśli element tekstowy nie ma elementów podrzędnych, ale ma atrybuty, zawartość tekstowa zostanie przekonwertowana na element z kluczem '#text', a zawartość będzie jego wartością; atrybuty zostaną przekonwertowane zgodnie z zasadą serializacji 1.

Przykład:

Wejście:

```
<xml>
  <foo bar="BAR">
    BAZ
  </foo>
</xml>
```

Wyjście:

```
{
  "xml": {
    "foo": {
      "@bar": "BAR",
      "#text": "BAZ"
    }
  }
}
```

Globalne funkcje JavaScript

Dodatkowe globalne funkcje JavaScript zostały zaimplementowane przy użyciu Duktape:

- `btoa(data)` - koduje dane do ciągu base64.
- `atob(base64_string)` - dekoduje ciąg base64 jako bufor Uint8Array.

```
try {
    b64 = btoa("test string");
    buffer = atob(b64);

    // Należy pamiętać, że logika dekodowania zależy od formatu danych w buforze.
    decoded = String.fromCharCode.apply(this, [].slice.call(buffer));
}
catch (error) {
    return {'error.name' : error.name, 'error.message' : error.message};
}
```

- `md5(data)` - oblicza skrót MD5 danych.
- `sha256(data)` - oblicza skrót SHA256 danych.

- `hmac('<hash type>',key,data)` - zwraca skrót HMAC jako ciąg w formacie szesnastkowym; jako `hash type` obsługiwane są `md5` i `sha256`; parametry `key` i `data` obsługują dane binarne.

  Przykłady:

  - `hmac('md5',key,data)`
  - `hmac('sha256',key,data)`

- `sign(hash,key,data)` - zwraca obliczony podpis (podpis RSA z SHA-256) jako ciąg, gdzie:
 hash - dozwolone jest tylko `sha256`, w przeciwnym razie zostanie zgłoszony błąd.
 key - klucz prywatny. Powinien być zgodny ze standardem PKCS#1 lub PKCS#8. Klucz może być podany w różnych formach:

  - ze spacjami zamiast znaków nowej linii
  - z poprzedzonymi znakiem ucieczki lub niepoprzedzonymi znakiem ucieczki `\n` zamiast znaków nowej linii
  - bez żadnych znaków nowej linii, jako ciąg jednoliniowy
  - jako ciąg w formacie JSON

  Klucz może być również wczytany z makra użytkownika/tajnego makra/sejfu.

  data - dane, które zostaną podpisane. Mogą to być ciąg znaków (obsługiwane są również dane binarne) lub bufor (Uint8Array/ArrayBuffer).

  Przykład:

  - `sign('sha256',key,data)`

  Do obliczania podpisów używany jest OpenSSL lub GnuTLS. Jeśli Zabbix został zbudowany bez którejkolwiek z tych bibliotek szyfrowania, zostanie zgłoszony błąd ('missing OpenSSL or GnuTLS library').

Obiekty JavaScript pozycji przeglądarki

Przegląd

Ta sekcja opisuje rozszerzenia języka JavaScript w Zabbix zaimplementowane przy użyciu Duktape do użycia w skrypcie pozycji Browser. Rozszerzenia te uzupełniają obiekty JavaScript opisane na stronie Dodatkowe obiekty JavaScript.

Browser

Obiekt `Browser` zarządza sesjami WebDriver, inicjalizując sesję podczas tworzenia i kończąc ją podczas niszczenia obiektu. Pojedynczy skrypt może obsługiwać maksymalnie cztery obiekty `Browser`.

Aby utworzyć obiekt `Browser`, użyj składni `new Browser(options)`. Parametr `options` (*obiekt JSON*) określa opcje przeglądarki, zwykle wynik metody opcji WebDriver (na przykład `Browser.chromeOptions()`).

Obiekt `Browser` obsługuje następujące metody.

Method	Description
`navigate(url)`	Przejdź do określonego adresu URL. Parametry: `url` - (string) Adres URL, do którego należy przejść.
`getUrl()`	Zwraca ciąg znaków z adresem URL otwartej strony.
`getPageSource()`	Zwraca ciąg znaków z kodem źródłowym otwartej strony.
`findElement(strategy, selector)`	Zwraca obiekt `Element` z jednym elementem na otwartej stronie (lub zwraca `null`, jeśli żadne elementy nie pasują do `strategy` i `selector`). Parametry: `strategy` - (string, css selector/link text/partial link text/tag name/xpath) Strategia lokalizacji; `selector` - (string) Selektor elementu używający określonej strategii lokalizacji.
`findElements(strategy, target)`	Zwraca tablicę obiektów `Element` z wieloma elementami na otwartej stronie (lub zwraca pustą tablicę, jeśli żadne elementy nie pasują do strategii lokalizacji i celu). Parametry: `strategy` - (string, css selector/link text/partial link text/tag name/xpath) Strategia lokalizacji; `target` - (string) Selektor elementu używający określonej strategii lokalizacji.
`getCookies()`	Zwraca tablicę obiektów `Cookie`.

Method	Description
`addCookie(cookie)`	Ustawia pliki cookie. Parametry: `cookie` - (`Cookie` object) Plik cookie do ustawienia.
`getScreenshot()`	Zwraca ciąg znaków zakodowany w base64, reprezentujący obraz obszaru widoku przeglądarki.
`setScreenSize(x,y)`	Ustawia rozmiar obszaru widoku przeglądarki. Parametry: `x` - (string) Szerokość w pikselach; `y` - (string) Wysokość w pikselach.
`setScriptTimeout(timeout)`	Ustawia limit czasu ładowania skryptu. Parametry: `timeout` - (integer) Wartość limitu czasu w milisekundach.
`setSessionTimeout(timeout)`	Ustawia limit czasu sesji (ładowania strony). Parametry: `timeout` - (integer) Wartość limitu czasu w milisekundach.
`setElementWaitTimeout(timeout)`	Ustawia limit czasu strategii lokalizacji elementu (implicit). Parametry: `timeout` - (integer) Wartość limitu czasu w milisekundach.
`collectPerfEntries(mark)`	Zbiera wpisy wydajności do pobrania metodą `getResult()`. Parametry: `mark` - (string, optional) Znacznik migawki wydajności.
`getRawPerfEntries()`	Zwraca tablicę obiektów wpisów wydajności.
`getResult()`	Zwraca obiekt `Result` ze statystykami sesji przeglądarki (informacje o błędach, migawki wydajności itp.).
`getError()`	Zwraca obiekt `BrowserError` z błędami przeglądarki (lub zwraca `null`, jeśli nie ma błędów przeglądarki).
`setError(message)`	Ustawia niestandardowy komunikat błędu, który ma zostać uwzględniony w obiekcie `Result`. Parametry: `message` - (string) Komunikat błędu.
`discardError()`	Odrzuca błąd, który ma zostać zwrócony w obiekcie `Result`.
`getAlert()`	Zwraca obiekt `Alert` z alertami przeglądarki (lub zwraca `null`, jeśli nie ma alertów przeglądarki).
`chromeOptions()`	Zwraca obiekt `chromeOptions` ze wstępnie zdefiniowanymi opcjami przeglądarki Chrome.
`firefoxOptions()`	Zwraca obiekt `firefoxOptions` ze wstępnie zdefiniowanymi opcjami przeglądarki Firefox.
`safariOptions()`	Zwraca obiekt `safariOptions` ze wstępnie zdefiniowanymi opcjami przeglądarki Safari.
`edgeOptions()`	Zwraca obiekt `edgeOptions` ze wstępnie zdefiniowanymi opcjami przeglądarki Edge.
`switchFrame(target)`	Przełącza do określonej ramki. Parametry: `target` - (browser element or integer, optional) Ramka docelowa. Aby wybrać ramkę według elementu, przekaż element. Aby wybrać ramkę według indeksu, przekaż liczbę. Jeśli pozostawiono puste, nastąpi przełączenie do kontekstu przeglądania najwyższego poziomu.

Wszystkie metody Browser mogą zgłaszać następujące błędy:

- `BrowserError` - dziedziczony z obiektu `Error`, zgłaszany, jeśli konstruktor `Browser` zakończy się niepowodzeniem; zawiera dodatkową właściwość `browser` z obiektem `Browser`, który zgłosił ten `BrowserError`.
- `WebdriverError` - dziedziczony z `BrowserError`; zawiera te same właściwości co obiekt `BrowserError`, które wskazują, czy błąd został wygenerowany w odpowiedzi na błąd w odpowiedzi WebDriver.

Element

Obiekt `Element` jest zwracany przez metody `findElement()`/`findElements()` obiektu `Browser` i nie może być tworzony bezpośrednio.

Obiekt `Element` reprezentuje element na stronie internetowej i udostępnia metody do interakcji z nim.

Następujące metody są obsługiwane dla obiektu `Element`.

Method	Description
`getAttribute(name)`	Zwraca ciąg znaków wartości atrybutu elementu (lub zwraca `null`, jeśli określony atrybut nie został znaleziony). Parameters: `name` - (string) Nazwa atrybutu.
`getProperty(name)`	Zwraca ciąg znaków wartości właściwości elementu (lub zwraca `null`, jeśli określona właściwość nie została znaleziona). Parameters: `name` - (string) Nazwa właściwości.
`getText()`	Zwraca ciąg znaków wartości tekstu elementu.
`click()`	Kliknij element.
`clear()`	Czyści zawartość edytowalnego elementu.
`sendKeys(keys)`	Wysyła klawisze. Parameters: `keys` - (string) Klawisze do wysłania.

Cookie

Obiekt `Cookie` jest zwracany przez metodę `getCookies()` obiektu `Browser` i przekazywany do metody `addCookie()`.

Chociaż obiekt `Cookie` nie ma żadnych metod, może zawierać następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa pliku cookie.
value	string	Wartość pliku cookie.
path	string	Ścieżka, dla której plik cookie jest prawidłowy. Domyślnie `"/"`, jeśli zostanie pominięta podczas dodawania pliku cookie.
domain	string	Domena, dla której plik cookie jest widoczny. Domyślnie domena URL aktywnego dokumentu bieżącego kontekstu przeglądania sesji, jeśli zostanie pominięta podczas dodawania pliku cookie.
secure	boolean	Wskazuje, czy plik cookie jest bezpieczny. Domyślnie `false`, jeśli zostanie pominięte podczas dodawania pliku cookie.
httpOnly	boolean	Wskazuje, czy plik cookie jest dostępny tylko przez HTTP. Domyślnie `false`, jeśli zostanie pominięte podczas dodawania pliku cookie.
expiry	integer	Czas wygaśnięcia pliku cookie (w sekundach od epoki Unix). Nie może być ustawiony, jeśli zostanie pominięty podczas dodawania pliku cookie.
sameSite	string	Atrybut `sameSite` pliku cookie, który określa, czy plik cookie powinien być ograniczony do kontekstu first-party lub same-site. Może być ustawiony na `"Lax"` lub `"Strict"`. Domyślnie `"None"`, jeśli zostanie pominięty podczas dodawania pliku cookie.

Alert

Obiekt `Alert` reprezentuje alert strony internetowej, jest zwracany przez metodę `getAlert()` obiektu `Browser` i nie może być tworzony bezpośrednio.

Obiekt `Alert` zawiera właściwość `text` z tekstem alertu (lub `null`, jeśli nie ma alertów).

Następujące metody są obsługiwane przez obiekt `Alert`.

Method	Description
`accept()`	Akceptuje alert.
`dismiss()`	Odrzuca alert.

Wynik

Obiekt `Result` zawiera statystyki sesji i jest zwracany przez metodę `getResult()` obiektu `Browser`.

Zazwyczaj obiekt `Result` jest konwertowany do postaci ciągu znaków i zwracany ze skryptu, a następnie przetwarzany na wartości zależnych pozycji za pomocą preprocessing.

Chociaż obiekt `Result` nie ma żadnych metod, może zawierać następujące właściwości.

Właściwość	Typ	Opis
duration	string	Czas trwania sesji od utworzenia sesji do pobrania wyniku.
error	object	Informacje o błędzie.
http_status	integer	Status HTTP zwrócony przez WebDriver (lub 0, jeśli nie ma błędów WebDriver).
error_code	string	Błąd zwrócony przez WebDriver (lub pusty ciąg znaków, jeśli nie ma błędów WebDriver).
message	string	Komunikat błędu WebDriver (lub pusty ciąg znaków, jeśli nie ma błędów WebDriver).
performance_data	object	Statystyki wydajności.
summary	object	Podsumowanie wydajności.
navigation	object	Podsumowanie nawigacji.
resource	object	Podsumowanie zasobów.
details	array of objects	Statystyki wydajności po każdej operacji, która mogła skutkować nawigacją.
mark	string	*(opcjonalnie)* Oznaczenie migawki wydajności określone za pomocą metody `collectPerfEntries()`.
navigation	object	Statystyki nawigacji.
resource	object	Podsumowanie zasobów dla tego kroku.
user	array of objects	Tablica statystyk typu mark/measure.
marks	array of objects	Indeksy oznaczonych migawek wydajności.
name	string	Nazwa oznaczenia migawki wydajności.
index	integer	Indeks migawki wydajności w tablicy details.

Wstępne przetwarzanie CSV do JSON

Przegląd

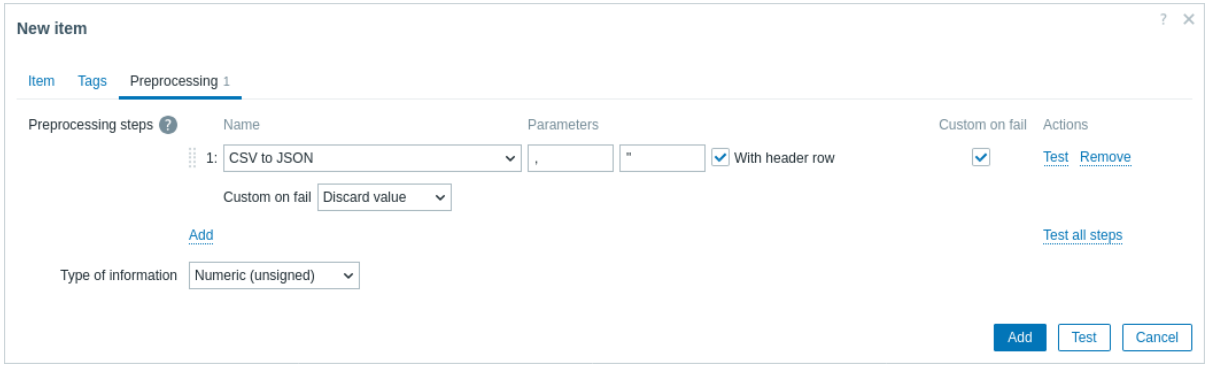
W tym kroku przetwarzania wstępnego można przekonwertować dane z pliku CSV do formatu JSON. Jest on obsługiwany w:

- pozycjach (prototypach pozycji)
- regułach wykrywania niskopoziomowego

Konfiguracja

Aby skonfigurować krok przetwarzania wstępnego CSV do JSON:

1. Przejdź do karty *Przetwarzanie wstępne* w konfiguracji pozycji/reguły wykrywania.
2. Kliknij *Dodaj*.
3. Wybierz opcję *CSV do JSON*.



Pierwszy parametr umożliwia ustawienie niestandardowego separatora. Należy pamiętać, że jeśli pierwszy wiersz danych wejściowych CSV zaczyna się od „Sep=" i następuje po nim pojedynczy znak UTF-8, znak ten zostanie użyty jako separator, jeśli pierwszy parametr nie został ustawiony. Jeśli pierwszy parametr nie został ustawiony, a separator nie został odczytany z wiersza „Sep=", jako separator zostanie użyty przecinek.

Drugi opcjonalny parametr umożliwia ustawienie symbolu cudzysłowu.

Jeśli pole wyboru *Z wierszem nagłówka* jest zaznaczone, wartości wiersza nagłówka zostaną zinterpretowane jako nazwy kolumn (więcej informacji można znaleźć w sekcji Przetwarzanie nagłówka).

Jeśli pole wyboru *Niestandardowe działanie w przypadku błędu* jest zaznaczone, pozycja nie otrzyma statusu nieobsługiwanej w przypadku nieudanego kroku przetwarzania wstępnego. Dodatkowo można ustawić niestandardowe opcje obsługi błędów: odrzucenie wartości, ustawienie określonej wartości lub ustawienie określonego komunikatu o błędzie.

Header processing

The CSV file header line can be processed in two different ways:

- If the *With header row* checkbox is marked - header line values are interpreted as column names. In this case the column names must be unique and the data row should not contain more columns than the header row;
- If the *With header row* checkbox is not marked - the header line is interpreted as data. Column names are generated automatically (1,2,3,4...)

CSV file example:

```
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:
A quotation character within a quoted field in the input must be escaped by preceding it with another quotation character.

Processing header line

JSON output when a header line is expected:

```
[
   {
      "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"
```

```
    }
]
```

No header line processing

JSON output when a header line is not expected:

```
[
    {
        "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"
    }
]
```

Typy pozycji

Przegląd

Typy pozycji obejmują różne metody pozyskiwania danych z systemu. Każdy typ pozycji ma własny zestaw obsługiwanych kluczy pozycji i wymaganych parametrów.

Zabbix oferuje obecnie następujące typy pozycji:

- Zabbix agent
  - Monitorowanie plików dziennika
- Proste sprawdzenie
  - Monitorowanie VMware
- SNMP agent
- Pułapka SNMP
- Wewnętrzny Zabbix
- Zabbix trapper
- Sprawdzenie zewnętrzne
- Monitorowanie bazy danych
- HTTP agent
  - Sprawdzenie Prometheus
- IPMI agent
- SSH agent
- Telnet agent
- JMX agent
- Pozycja obliczana
- Pozycja zależna
- Pozycja skryptowa
- Pozycja przeglądarki
- Zapytanie telemetryczne

Szczegółowe informacje dotyczące wszystkich typów pozycji zawarto na podstronach tej sekcji. Chociaż typy pozycji oferują wiele opcji gromadzenia danych, dostępne są także dodatkowe opcje za pośrednictwem parametrów użytkownika lub modułów ładowalnych.

Niektóre sprawdzenia są wykonywane wyłącznie przez serwer Zabbix (jako monitorowanie bez agenta), podczas gdy inne wymagają agenta Zabbix lub nawet Java gateway Zabbix (w przypadku monitorowania JMX).

Attention:
Jeśli określony typ pozycji wymaga określonego interfejsu (na przykład sprawdzenie IPMI wymaga interfejsu IPMI na hoście), ten interfejs musi istnieć w definicji hosta.

W definicji hosta można skonfigurować wiele interfejsów: Zabbix agent, SNMP agent, JMX i IPMI. Jeśli pozycja może korzystać z więcej niż jednego interfejsu, przeszuka dostępne interfejsy hosta (w kolejności: Agent→SNMP→JMX→IPMI) i wybierze pierwszy odpowiedni interfejs, z którym zostanie powiązana.

Wszystkie pozycje, które zwracają tekst (informacje typu znakowego, dziennika lub tekstowego), mogą również zwracać wyłącznie białe znaki (jeśli ma to zastosowanie), ustawiając wartość zwracaną na pusty ciąg znaków (obsługiwane od wersji 2.0).

Agent Zabbix

Przegląd

Ta sekcja zawiera szczegółowe informacje o kluczach pozycji, które wykorzystują komunikację z agentem Zabbix do zbierania danych.

Istnieją pasywne i aktywne sprawdzenia agenta. Podczas konfigurowania pozycji można wybrać wymagany typ:

- *Zabbix agent* - dla sprawdzeń pasywnych
- *Zabbix agent (active)* - dla sprawdzeń aktywnych

Należy pamiętać, że wszystkie klucze pozycji obsługiwane przez agenta Zabbix są również obsługiwane przez nową generację agenta Zabbix 2. Zobacz dodatkowe klucze pozycji dostępne wyłącznie dla agenta 2.

Obsługiwane klucze pozycji

Poniżej wymieniono klucze pozycji, których można używać z agentem Zabbix.

Klucze pozycji są wymienione bez parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby wyświetlić pełne informacje.

Klucz pozycji	Opis	Grupa pozycji
kernel.maxfiles	Maksymalna liczba otwartych plików obsługiwana przez system operacyjny.	Jądro
kernel.maxproc	Maksymalna liczba procesów obsługiwana przez system operacyjny.	
kernel.openfiles	Liczba aktualnie otwartych deskryptorów plików.	
log	Monitorowanie pliku dziennika.	Monitorowanie dzienników
log.count	Liczba pasujących wierszy w monitorowanym pliku dziennika.	
logrt	Monitorowanie rotowanego pliku dziennika.	
logrt.count	Liczba pasujących wierszy w monitorowanym rotowanym pliku dziennika.	
modbus.get	Odczytuje dane Modbus.	Modbus
net.dns	Sprawdza stan usługi DNS.	Sieć
net.dns.perf	Sprawdza wydajność usługi DNS.	
net.dns.record	Wykonuje zapytanie DNS.	
net.if.collisions	Liczba kolizji poza oknem.	
net.if.discovery	Lista interfejsów sieciowych.	
net.if.get	Lista interfejsów sieciowych ze szczegółowymi informacjami.	
net.if.in	Statystyki ruchu przychodzącego na interfejsie sieciowym.	
net.if.out	Statystyki ruchu wychodzącego na interfejsie sieciowym.	
net.if.total	Suma statystyk ruchu przychodzącego i wychodzącego na interfejsie sieciowym.	
net.tcp.listen	Sprawdza, czy ten port TCP jest w stanie LISTEN.	
net.tcp.port	Sprawdza, czy można nawiązać połączenie TCP z określonym portem.	
net.tcp.service	Sprawdza, czy usługa działa i akceptuje połączenia TCP.	
net.tcp.service.perf	Sprawdza wydajność usługi TCP.	
net.tcp.socket.count	Zwraca liczbę gniazd TCP pasujących do parametrów.	
net.udp.listen	Sprawdza, czy ten port UDP jest w stanie LISTEN.	
net.udp.service	Sprawdza, czy usługa działa i odpowiada na żądania UDP.	

Klucz pozycji	Opis	Grupa pozycji
net.udp.service.perf	Sprawdza wydajność usługi UDP.	
net.udp.socket.count	Zwraca liczbę gniazd UDP pasujących do parametrów.	
proc.cpu.util	Procent wykorzystania procesora przez proces.	Procesy
proc.get	Lista procesów systemu operacyjnego i ich parametrów.	
proc.mem	Pamięć używana przez proces, wyrażona w bajtach.	
proc.num	Liczba procesów.	
sensor	Odczyt z czujnika sprzętowego.	Czujniki
system.boottime	Czas uruchomienia systemu.	System
system.cpu.discovery	Lista wykrytych procesorów lub rdzeni procesora.	
system.cpu.intr	Przerwania urządzeń.	
system.cpu.load	Obciążenie procesora.	
system.cpu.num	Liczba procesorów.	
system.cpu.switches	Liczba przełączeń kontekstu.	
system.cpu.util	Procent wykorzystania procesora.	
system.hostname	Nazwa hosta systemu.	
system.hw.chassis	Informacje o obudowie.	
system.hw.cpu	Informacje o procesorze.	
system.hw.devices	Lista urządzeń PCI lub USB.	
system.hw.macaddr	Lista adresów MAC.	
system.localtime	Czas systemowy.	
system.run	Uruchamia określone polecenie na hoście.	
system.stat	Statystyki systemowe.	
system.sw.arch	Informacje o architekturze oprogramowania.	
system.sw.os	Informacje o systemie operacyjnym.	
system.sw.os.get	Szczegółowe informacje o systemie operacyjnym (wersja, typ, nazwa dystrybucji, wersja główna i pomocnicza itd.).	
system.sw.packages	Lista zainstalowanych pakietów.	
system.sw.packages.get	Szczegółowa lista zainstalowanych pakietów.	
system.swap.in	Statystyki wczytywania do obszaru wymiany (z urządzenia do pamięci).	
system.swap.out	Statystyki zapisywania do obszaru wymiany (z pamięci na urządzenie).	
system.swap.size	Rozmiar obszaru wymiany w bajtach lub jako procent całkowitego rozmiaru.	
system.uname	Identyfikacja systemu.	
system.uptime	Czas działania systemu w sekundach.	
system.users.num	Liczba zalogowanych użytkowników.	
vfs.dev.discovery	Lista urządzeń blokowych i ich typów.	Wirtualne systemy plików
vfs.dev.get	Lista urządzeń blokowych ze szczegółowymi informacjami.	
vfs.dev.read	Statystyki odczytu dysku.	
vfs.dev.write	Statystyki zapisu na dysku.	
vfs.dir.count	Liczba wpisów w katalogu.	
vfs.dir.get	Lista wpisów w katalogu.	
vfs.dir.size	Rozmiar katalogu.	
vfs.file.cksum	Suma kontrolna pliku obliczona za pomocą algorytmu UNIX cksum.	
vfs.file.contents	Pobieranie zawartości pliku.	
vfs.file.exists	Sprawdza, czy plik istnieje.	
vfs.file.get	Zwraca informacje o pliku.	
vfs.file.md5sum	Suma kontrolna MD5 pliku.	
vfs.file.owner	Pobiera właściciela pliku.	
vfs.file.permissions	Zwraca 4-cyfrowy ciąg zawierający liczbę ósemkową z uprawnieniami UNIX.	
vfs.file.regexp	Pobiera ciąg znaków z pliku.	
vfs.file.regmatch	Znajduje ciąg znaków w pliku.	
vfs.file.size	Rozmiar pliku.	
vfs.file.time	Informacje o czasie pliku.	
vfs.fs.discovery	Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania.	
vfs.fs.get	Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania, a w trybie *full* także z dostępnym miejscem na dysku i statystykami i-węzłów.	
vfs.fs.inode	Liczba lub procent i-węzłów.	
vfs.fs.size	Miejsce na dysku w bajtach lub jako procent całkowitego rozmiaru.	
vm.memory.size	Rozmiar pamięci w bajtach lub jako procent całkowitego rozmiaru.	Pamięć wirtualna

Klucz pozycji	Opis	Grupa pozycji
web.page.get	Pobiera zawartość strony internetowej.	Monitorowanie sieci Web
web.page.perf	Czas ładowania całej strony internetowej.	
web.page.regexp	Znajduje ciąg znaków na stronie internetowej.	
agent.hostmetadata	Metadane hosta agenta.	Zabbix
agent.hostname	Nazwa hosta agenta.	
agent.ping	Sprawdzenie dostępności agenta.	
agent.variant	Wariant agenta Zabbix (agent Zabbix lub agent Zabbix 2).	
agent.version	Wersja agenta Zabbix.	
zabbix.stats	Zwraca zestaw wewnętrznych metryk serwera Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.	
zabbix.stats (queue)	Zwraca liczbę monitorowanych pozycji w kolejce, których przetwarzanie na serwerze Zabbix lub proxy jest opóźnione. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.	

Obsługiwane platformy

O ile w szczegółach pozycji nie określono inaczej, pozycje agenta (oraz wszystkie parametry) są obsługiwane na:

- **Linux**
- **FreeBSD**
- **Solaris**
- **HP-UX**
- **AIX**
- **MacOS X**
- **OpenBSD**
- **NetBSD**

Wiele pozycji agenta jest również obsługiwanych w systemie **Windows**. Szczegóły można znaleźć na stronie Windows agent item.

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

kernel.maxfiles

 Maksymalna liczba otwartych plików obsługiwana przez system operacyjny.
 Zwracana wartość: *Liczba całkowita*.
 Obsługiwane platformy: Linux, FreeBSD, MacOS X, OpenBSD, NetBSD.

kernel.maxproc

 Maksymalna liczba procesów obsługiwana przez system operacyjny.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux 2.6 i nowsze, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

kernel.openfiles

 Liczba aktualnie otwartych deskryptorów plików.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux (pozycja może działać na innych platformach typu UNIX).

log[file,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent dir>]

 Monitorowanie pliku dziennika.
 Wartość zwracana: *Log*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka i nazwa pliku dziennika;
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej identifier;
- **maxlines** - maksymalna liczba nowych linii na sekundę, jaką agent wyśle do serwera Zabbix lub proxy. Ten parametr zastępuje wartość `MaxLinesPerSecond` w `zabbix_agentd.conf`;
- **mode** - możliwe wartości:
  - *all* (domyślnie);
  - *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji);
- **output** - opcjonalny szablon formatowania wyjścia. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, w którym zaczyna się dopasowanie, do znaku, w którym się kończy), natomiast sekwencja **\N**

(gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem, jeśli N przekracza liczbę przechwyconych grup);
- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości:
  - 0 - (domyślnie) nigdy nie ignoruj linii pliku dziennika;
  - > 0.0 - ignoruj starsze linie, aby najnowsze analizowane linie mieściły się w czasie ”maxdelay” sekund. Przed użyciem zapoznaj się z uwagami dotyczącymi maxdelay!
- **options** - dodatkowe opcje:
mtime-noreread - rekordy nieunikalne, ponowne odczytanie tylko wtedy, gdy zmieni się rozmiar pliku (ignoruj zmianę czasu modyfikacji). (Ten parametr jest przestarzały od wersji 5.0.2, ponieważ obecnie mtime jest ignorowany.)
- **persistent dir** (tylko w zabbix_agentd w systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące plików trwałych.

Uwagi:

- Pozycja musi być skonfigurowana jako active check.
- Jeśli plik jest niedostępny lub uprawnienia nie pozwalają na dostęp, pozycja staje się nieobsługiwana.
- Jeśli `output` pozostanie puste - zwracana jest cała linia zawierająca dopasowany tekst. Należy pamiętać, że wszystkie globalne typy wyrażeń regularnych z wyjątkiem *Matches regular expression* zawsze zwracają całą dopasowaną linię, a parametr `output` jest ignorowany.
- Ekstrakcja zawartości z użyciem parametru `output` odbywa się na agencie.

Przykłady:

```
log[/var/log/syslog]
log[/var/log/syslog,error]
log[/home/zabbix/logs/logfile,,,100]
```

Przykład użycia parametru `output` do wyodrębnienia liczby z rekordu dziennika:

```
log[/app1/app.log,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #ta pozycja dop
```

Przykład użycia parametru `output` do przekształcenia rekordu dziennika przed wysłaniem do serwera:

```
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>]

 Liczba dopasowanych linii w monitorowanym pliku dziennika.
 Wartość zwracana: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka i nazwa pliku dziennika;
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej identifier;
- **maxproclines** - maksymalna liczba nowych linii na sekundę, które agent będzie analizował (nie może przekroczyć 10000). Wartość domyślna to 10*`MaxLinesPerSecond` w `zabbix_agentd.conf`.
- **mode** - możliwe wartości:
  - *all* (domyślnie);
  - *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).
- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości:
  - 0 - (domyślnie) nigdy nie ignoruj linii pliku dziennika;
  - > 0.0 - ignoruj starsze linie, aby najnowsze linie zostały przeanalizowane w ciągu ”maxdelay” sekund. Przed użyciem zapoznaj się z uwagami dotyczącymi maxdelay!
- **options** - dodatkowe opcje:
mtime-noreread - rekordy nieunikalne, ponowne odczytanie tylko wtedy, gdy rozmiar pliku się zmieni (ignoruj zmianę czasu modyfikacji). (Ten parametr jest przestarzały od wersji 5.0.2, ponieważ obecnie mtime jest ignorowany.)
- **persistent dir** (tylko w `zabbix_agentd` na systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka do katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące plików trwałych.

Uwagi:

- Pozycja musi być skonfigurowana jako active check.
- Dopasowane linie są zliczane w nowych liniach od ostatniego sprawdzenia logu przez agenta, a więc zależą od interwału aktualizacji pozycji.
- Jeśli plik jest niedostępny lub uprawnienia nie pozwalają na dostęp, pozycja staje się nieobsługiwana.

logrt[file regexp,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent dir>]

 Monitorowanie obracanego pliku dziennika.
 Wartość zwracana: *Log*.
 Zobacz obsługiwane platformy.

Parametry:

- **file regexp** - bezwzględna ścieżka do pliku, z nazwą pliku określoną za pomocą wyrażenia regularnego expression. Należy pamiętać, że wyrażenie regularne ma zastosowanie tylko do nazwy pliku i nie musi dopasowywać całej nazwy (np. `/path/to/agent` dopasuje `zabbix_agentd.log`).
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec zawartości.
- **encoding** - identyfikator strony kodowej identifier.
- **maxlines** - maksymalna liczba nowych linii na sekundę, które agent wyśle do serwer Zabbix lub proxy. Ten parametr zastępuje wartość `MaxLinesPerSecond` w `zabbix_agentd.conf`.
- **mode** - możliwe wartości:
  - *all* (domyślnie);
  - *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).
- **output** - opcjonalny szablon formatowania wyjścia. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, w którym zaczyna się dopasowanie, do znaku, w którym dopasowanie się kończy), natomiast sekwencja **\N** (gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem, jeśli N przekracza liczbę przechwyconych grup).
- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości:
  - 0 - (domyślnie) nigdy nie ignoruj linii pliku dziennika;
  - > 0.0 - ignoruj starsze linie, aby najnowsze analizowane linie zostały przetworzone w ciągu ”maxdelay” sekund. Przed użyciem przeczytaj uwagi dotyczące maxdelay!
- **options** - typ rotacji pliku dziennika i inne opcje. Możliwe wartości:
  - *rotate* (domyślnie).
  - *copytruncate* - należy pamiętać, że *copytruncate* nie może być używane razem z *maxdelay*. W takim przypadku *maxdelay* musi mieć wartość 0 lub nie może być określony; zobacz uwagi dotyczące copytruncate.
  - *mtime-reread* - rekordy nieunikalne, odczytaj ponownie, jeśli zmieni się czas modyfikacji lub rozmiar (używane domyślnie, jeśli żaden `mtime-*` nie jest jawnie ustawiony).
  - *mtime-noreread* - rekordy nieunikalne, odczytaj ponownie tylko jeśli zmieni się rozmiar (ignoruj zmianę czasu modyfikacji).
- **persistent dir** (tylko w `zabbix_agentd` w systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka do katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące persistent files.

Uwagi:

- Pozycja musi być skonfigurowana jako active check.
- Rotacja plików dziennika opiera się na czasie ostatniej modyfikacji plików.
- Należy pamiętać, że `logrt` jest przeznaczony do pracy z jednym aktualnie aktywnym plikiem dziennika, przy czym kilka innych pasujących nieaktywnych plików jest rotowanych. Jeśli na przykład katalog zawiera wiele aktywnych plików dziennika, dla każdego z nich należy utworzyć osobną pozycję `logrt`. W przeciwnym razie, jeśli jedna pozycja `logrt` obejmie zbyt wiele plików, może to doprowadzić do wyczerpania pamięci i awarii monitorowania.
- Jeśli `output` pozostanie puste - zwracana jest cała linia zawierająca dopasowany tekst. Należy pamiętać, że wszystkie globalne typy wyrażeń regularnych z wyjątkiem *Matches regular expression* zawsze zwracają całą dopasowaną linię, a parametr `output` jest ignorowany.
- Wyodrębnianie zawartości przy użyciu parametru `output` odbywa się na agencie.
- W parametrze `file regexp` ścieżka do katalogu dziennika i wyrażenie regularne pliku dziennika muszą być rozdzielone poprawnym separatorem katalogów:
  - W systemie Windows separatorem musi być ukośnik odwrotny (\); ukośniki ukośne mogą być tolerowane w innych miejscach, z wyjątkiem tego, który oddziela ścieżkę katalogu dziennika od wyrażenia regularnego pliku dziennika (zobacz przykłady poniżej).
  - W innych systemach musi to być ukośnik ukośny (/).

Przykłady dla 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
```

Przykłady dla innych systemów:

```
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
```

Przykład użycia parametru `output` do wyodrębnienia liczby z rekordu dziennika:

```
logrt[/app1/^test.*log$,"task run [0-9.]+ sec, processed ([0-9]+) records, [0-9]+ errors",,,,\1] #this ite
```

Przykład użycia parametru `output` do przepisania rekordu dziennika przed wysłaniem do serwer:

```
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>]

 Liczba dopasowanych linii w monitorowanym pliku dziennika, który jest rotowany.
 Wartość zwracana: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **file regexp** - bezwzględna ścieżka do pliku, z nazwą pliku określoną za pomocą regularnego wyrażenia. Należy pamiętać, że wyrażenie regularne ma zastosowanie tylko do nazwy pliku i nie musi dopasowywać całej nazwy (np. `/path/to/agent` dopasuje `zabbix_agentd.log`).
- **regexp** - regularne wyrażenie opisujące wymagany wzorzec.
- **encoding** - identyfikator strony kodowej identifier.
- **maxproclines** - maksymalna liczba nowych linii na sekundę, które agent będzie analizował (nie może przekroczyć 10000). Wartość domyślna to 10*`MaxLinesPerSecond` w `zabbix_agentd.conf`.
- **mode** - możliwe wartości:
  - *all* (domyślnie);
  - *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).
- **maxdelay** - maksymalne opóźnienie w sekundach. Typ: float. Wartości:
  - 0 - (domyślnie) nigdy nie ignoruj linii pliku dziennika;
  - > 0.0 - ignoruj starsze linie, aby najnowsze linie zostały przeanalizowane w ciągu ”maxdelay” sekund. Przed użyciem przeczytaj uwagi dotyczące `maxdelay`!
- **options** - typ rotacji pliku dziennika i inne opcje. Możliwe wartości:
  - *rotate* (domyślnie).
  - *copytruncate* - należy pamiętać, że `copytruncate` nie może być używany razem z `maxdelay`. W takim przypadku `maxdelay` musi mieć wartość 0 lub nie może być określony; zobacz uwagi dotyczące `copytruncate`.
  - *mtime-reread* - rekordy nieunikalne, ponowne odczytanie, jeśli zmieni się czas modyfikacji lub rozmiar (domyślnie).
  - *mtime-noreread* - rekordy nieunikalne, ponowne odczytanie tylko jeśli zmieni się rozmiar (ignoruj zmianę czasu modyfikacji).
- **persistent dir** (tylko w `zabbix_agentd` na systemach Unix; nieobsługiwane w Zabbix agent 2) - bezwzględna ścieżka do katalogu, w którym mają być przechowywane pliki trwałe. Zobacz także dodatkowe uwagi dotyczące plików trwałych.

Uwagi:

- Pozycja musi być skonfigurowana jako aktywny check.
- Dopasowane linie są zliczane w nowych liniach od ostatniego sprawdzenia dziennika przez agenta, a więc zależą od interwału aktualizacji pozycji.
- Rotacja dziennika opiera się na czasie ostatniej modyfikacji plików.
- W parametrze `file regexp` ścieżka do katalogu dziennika i wyrażenie regularne pliku dziennika muszą być oddzielone poprawnym separatorem katalogów:
  - W systemie Windows separatorem musi być ukośnik odwrotny (\); ukośniki ukośne mogą być tolerowane w innych miejscach, z wyjątkiem tego, który oddziela ścieżkę do katalogu dziennika i wyrażenie regularne pliku dziennika (zobacz przykłady poniżej).
  - W innych systemach musi to być ukośnik ukośny (/).

Przykłady dla Windows:

```
logrt.count["c:/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #ta pozycja będzie zliczać liczbę dopa
logrt.count["//example.com/share/dir1/dir2/dir3\filename.*\.log","pattern_to_match"] #ta pozycja będzie zl
```

Przykłady dla innych systemów:

```
logrt.count["/home/zabbix/logs/.*","^logfile[0-9]{1,3}$",,100] #ta pozycja będzie zliczać liczbę dopasowań
logrt.count["/home/user/^logfile_.*_[0-9]{1,3}$","pattern_to_match","UTF-8",100] #ta pozycja będzie zlicza
```

modbus.get[endpoint,<slave id>,<function>,<address>,<count>,<type>,<endianness>,<offset>]

 Odczytuje dane Modbus.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux, Windows.

Parametry:

- **endpoint** - punkt końcowy zdefiniowany jako `protocol://connection_string`;
- **slave id** - identyfikator slave;
- **function** - funkcja Modbus;
- **address** - adres pierwszego rejestru, cewki lub wejścia;
- **count** - liczba rekordów do odczytu;
- **type** - typ danych;
- **endianness** - konfiguracja kolejności bajtów;

- **offset** - liczba rejestrów, począwszy od `address`, których wyniki zostaną odrzucone.

W systemie Windows ta pozycja podąża za ścieżkami UNC w parametrze `endpoint`, co może stanowić zagrożenie bezpieczeństwa.

Zobacz szczegółowy opis parametrów.

net.dns[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Sprawdza stan usługi DNS.
 Zwracane wartości: - 0 - rozpoznawanie DNS nie powiodło się (serwer DNS nie odpowiedział lub zwrócił błąd); - 1 - rozpoznawanie DNS powiodło się.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** (ignorowane w systemie Windows, chyba że używasz Zabbix agent 2) - adres IP serwera DNS (pozostaw puste, aby użyć domyślnego serwera DNS);
- **name** - nazwa DNS do zapytania;
- **type** - typ rekordu, o który ma zostać wykonane zapytanie (domyślnie `SOA`);
- **timeout** (ignorowane w systemie Windows, chyba że używasz Zabbix agent 2) - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** (ignorowane w systemie Windows, chyba że używasz Zabbix agent 2) - liczba prób dla żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*.

Uwagi:

- Możliwe wartości dla `type` to: *ANY*, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (nieobsługiwane przez Zabbix agent w systemie Windows, Zabbix agent 2 we wszystkich systemach operacyjnych), *HINFO*, *MINFO*, *TXT*, *SRV*.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` ma wartość *PTR*) można podać nazwę DNS zarówno w formacie odwróconym, jak i nieodwróconym (zobacz przykłady poniżej). Pamiętaj, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Zinternacjonalizowane nazwy domen nie są obsługiwane, użyj zamiast nich nazw zakodowanych w IDNA.

Przykłady:

```
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.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.0.f.0.0.4.0.5.4.1.0.0.a.2.ip6.arpa,PTR]
```

net.dns.perf[<ip>,name,<type>,<timeout>,<count>,<protocol>]

 Sprawdza wydajność usługi DNS.
 Wartość zwracana: *Float* (0 - usługa jest niedostępna; sekundy - liczba sekund spędzonych na oczekiwaniu na odpowiedź od usługi).
 Zobacz obsługiwane platformy.

Parametry:

- **ip** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - adres IP serwera DNS (pozostaw puste, aby użyć domyślnego serwera DNS);
- **name** - nazwa DNS do zapytania;
- **type** - typ rekordu, o który ma zostać wykonane zapytanie (domyślnie `SOA`);
- **timeout** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - liczba prób dla żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*.

Uwagi:

- Możliwe wartości dla `type` to:
ANY, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (nieobsługiwane dla Zabbix agent w systemie Windows, Zabbix agent 2 we wszystkich systemach operacyjnych), *HINFO*, *MINFO*, *TXT*, *SRV*.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` jest ustawiony na *PTR*) można podać nazwę DNS zarówno w formacie odwróconym, jak i nieodwróconym (zobacz przykłady poniżej). Należy pamiętać, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Zinternacjonalizowane nazwy domen nie są obsługiwane, należy zamiast tego używać nazw zakodowanych w IDNA.
- Pozycja zwraca czas odpowiedzi zamiast 0, gdy serwer DNS odpowiada kodem błędu (na przykład `NXDOMAIN` lub `SERVFAIL`).

Przykłady:

```
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>]

 Wykonuje zapytanie DNS.

Zabbix agent 2 udostępnia również net.dns.get, które oferuje dodatkowe funkcje, takie jak większa liczba typów rekordów i większa kontrola nad monitorowaniem DNS. Wartość zwracana: ciąg znaków z wymaganym typem informacji.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - adres IP serwera DNS (pozostaw puste, aby użyć domyślnego serwera DNS);
- **name** - nazwa DNS do zapytania;
- **type** - typ rekordu, o który ma zostać wykonane zapytanie (domyślnie SOA);
- **timeout** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** (ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2) - liczba prób dla żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*.

Uwagi:

- Możliwe wartości dla type to:
ANY, *A*, *NS*, *CNAME*, *MB*, *MG*, *MR*, *PTR*, *MD*, *MF*, *MX*, *SOA*, *NULL*, *WKS* (nieobsługiwane dla Zabbix agent w systemie Windows, Zabbix agent 2 we wszystkich systemach operacyjnych), *HINFO*, *MINFO*, *TXT*, *SRV*.
- W przypadku odwrotnych wyszukiwań DNS (gdy type jest ustawiony na *PTR*) można podać nazwę DNS w formacie odwróconym lub nieodwróconym (zobacz przykłady poniżej). Należy pamiętać, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Nazwy domen z internacjonalizacją nie są obsługiwane, należy zamiast tego używać nazw zakodowanych w IDNA.

Przykłady:

```
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]

 Liczba kolizji poza oknem czasowym.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD. Uprawnienia root są wymagane w NetBSD.

Parametr:

- **if** - nazwa interfejsu sieciowego

net.if.discovery

 Lista interfejsów sieciowych. Używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

net.if.get[<regexp>]

 Lista interfejsów sieciowych ze szczegółowymi informacjami.
 Używane do wykrywania niskiego poziomu.
 Ta pozycja jest rozszerzoną wersją pozycji Zabbix agent `net.if.discovery`.
 Wartość zwracana: *obiekt JSON*.
 Obsługiwane platformy: Linux.

Parametry:

- **regexp** - wyrażenie regularne expression dopasowujące nazwy interfejsów sieciowych.

Komentarze:

- Zwraca obiekt JSON z interfejsami sieciowymi, których nazwy pasują do podanego wyrażenia regularnego (zobacz więcej szczegółów).

net.if.in[if,<mode>]

 Statystyki ruchu przychodzącego na interfejsie sieciowym.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Na NetBSD wymagane są uprawnienia roota.

Parametry:

- **if** - nazwa interfejsu sieciowego (Unix); pełny opis interfejsu sieciowego lub adres IPv4; albo, jeśli ujęte w nawiasy klamrowe, GUID interfejsu sieciowego (Windows);
- **mode** - możliwe wartości:
  - *bytes* - liczba bajtów (domyślnie);
  - *packets* - liczba pakietów;
  - *errors* - liczba błędów;
  - *dropped* - liczba odrzuconych pakietów;
  - *overruns (fifo)* - liczba błędów bufora FIFO;
  - *frame* - liczba błędów ramek pakietów;
  - *compressed* - liczba skompresowanych pakietów odebranych przez sterownik urządzenia;
  - *multicast* - liczba ramek multicast odebranych przez sterownik urządzenia.

Uwagi:

- Możesz użyć tego klucza z krokiem przetwarzania *Change per second*, aby uzyskać statystyki w bajtach na sekundę.
- Tryb *dropped* jest obsługiwany tylko w Linux, FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Tryby *overruns*, *frame*, *compressed*, *multicast* są obsługiwane tylko w Linux.
- W HP-UX ta pozycja nie udostępnia szczegółów dla interfejsów loopback (np. lo0).

Przykłady:

```
net.if.in[eth0]
net.if.in[eth0,errors]
```

net.if.out[if,<mode>]

 Statystyki ruchu wychodzącego na interfejsie sieciowym.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Na NetBSD wymagane są uprawnienia roota.

Parametry:

- **if** - nazwa interfejsu sieciowego (Unix); pełny opis interfejsu sieciowego lub adres IPv4; albo, jeśli ujęte w nawiasy klamrowe, GUID interfejsu sieciowego (Windows);
- **mode** - możliwe wartości:
  - *bytes* - liczba bajtów (domyślnie);
  - *packets* - liczba pakietów;
  - *errors* - liczba błędów;
  - *dropped* - liczba odrzuconych pakietów;
  - *overruns (fifo)* - liczba błędów bufora FIFO;
  - *collisions (colls)* - liczba kolizji wykrytych na interfejsie;
  - *carrier* - liczba utrat nośnika wykrytych przez sterownik urządzenia;
  - *compressed* - liczba skompresowanych pakietów przesłanych przez sterownik urządzenia.

Uwagi:

- Możesz użyć tego klucza z krokiem przetwarzania *Change per second*, aby uzyskać statystyki w bajtach na sekundę.
- Tryb *dropped* jest obsługiwany tylko w Linux, HP-UX.
- Tryby *overruns*, *collision*, *carrier*, *compressed* są obsługiwane tylko w Linux.
- W HP-UX ta pozycja nie udostępnia szczegółów dla interfejsów loopback (np. lo0).

Przykłady:

```
net.if.out[eth0]
net.if.out[eth0,errors]
```

net.if.total[if,<mode>]

 Suma statystyk ruchu przychodzącego i wychodzącego na interfejsie sieciowym.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[5], HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows. Uprawnienia roota są wymagane w systemie NetBSD.

Parametry:

- **if** - nazwa interfejsu sieciowego (Unix); pełny opis interfejsu sieciowego lub adres IPv4; albo, jeśli ujęte w nawiasy klamrowe, GUID interfejsu sieciowego (Windows);

- **mode** - możliwe wartości:
  - *bytes* - liczba bajtów (domyślnie);
  - *packets* - liczba pakietów;
  - *errors* - liczba błędów;
  - *dropped* - liczba odrzuconych pakietów;
  - *overruns (fifo)* - liczba błędów bufora FIFO;
  - *collisions (colls)* - liczba kolizji wykrytych na interfejsie;
  - *compressed* - liczba skompresowanych pakietów wysłanych lub odebranych przez sterownik urządzenia.

Uwagi:

- Możesz użyć tego klucza z krokiem przetwarzania *Change per second*, aby uzyskać statystyki bajtów na sekundę.
- Tryb *dropped* jest obsługiwany tylko w systemach Linux, HP-UX. Odrzucone pakiety są obsługiwane tylko wtedy, gdy zarówno `net.if.in`, jak i `net.if.out` działają dla odrzuconych pakietów na Twojej platformie.
- Tryby *overruns*, *collision*, *compressed* są obsługiwane tylko w systemie Linux.
- W systemie HP-UX ta pozycja nie udostępnia szczegółów dla interfejsów loopback (np. lo0).

Przykłady:

```
net.if.total[eth0]
net.if.total[eth0,errors]
```

net.tcp.listen[port]

 Sprawdza, czy ten port TCP znajduje się w stanie LISTEN.
 Zwracane wartości: 0 - nie znajduje się w stanie LISTEN; 1 - znajduje się w stanie LISTEN.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, MacOS X, Windows.

Parametr:

- **port** - numer portu TCP

W jądrze Linux 2.6.14 i nowszych informacje o nasłuchujących gniazdach TCP są, jeśli to możliwe, pobierane z interfejsu NETLINK jądra. W przeciwnym razie informacje są pobierane z plików `/proc/net/tcp` i `/roc/net/tcp6`.

Przykład:

```
net.tcp.listen[80]
```

net.tcp.port[<ip>,port]

 Sprawdza, czy możliwe jest nawiązanie połączenia TCP z określonym portem.
 Zwracane wartości: 0 - nie można się połączyć; 1 - można się połączyć.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** - adres IP lub nazwa DNS (domyślnie `127.0.0.1`);
- **port** - numer portu.

Uwagi:

- Do prostego testowania wydajności TCP użyj `net.tcp.service.perf[tcp,<ip>,<port>]`.
- Te kontrole mogą powodować dodatkowe komunikaty w plikach dziennika demonów systemowych (zwykle rejestrowane są sesje SMTP i SSH).

Przykład:

```
net.tcp.port[,80] #ta pozycja może być użyta do testowania dostępności serwera WWW działającego na porcie
```

net.tcp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i akceptuje połączenia TCP.
 Zwracane wartości: 0 - usługa jest niedostępna; 1 - usługa działa.
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* lub *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie `127.0.0.1`);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Te kontrole mogą powodować dodatkowe komunikaty w plikach dziennika systemowych demonów (zwykle rejestrowane są sesje SMTP i SSH).
- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście proszę użyć `net.tcp.port[]` do takich kontroli.

- Sprawdzanie LDAP i HTTPS w systemie Windows jest obsługiwane tylko przez Zabbix agent 2.
- Kontrola telnet sprawdza, czy pojawia się monit logowania (':' na końcu).

Przykład:

```
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>]

 Sprawdza wydajność usługi TCP.
 Zwracane wartości: *Float* (0 - usługa jest niedostępna; sekundy - liczba sekund spędzonych na oczekiwaniu na odpowiedź od usługi).
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https* lub *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie `127.0.0.1`);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Sprawdzanie protokołów szyfrowanych (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście prosimy użyć `net.tcp.service.perf[tcp,<ip>,<port>]` do takich sprawdzeń.
- Sprawdzenie telnet wyszukuje monit logowania (':' na końcu).

Przykład:

```
net.tcp.service.perf[ssh] #ten item może być użyty do testowania szybkości początkowej odpowiedzi z serwer
```

net.tcp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>]

 Zwraca liczbę gniazd TCP pasujących do parametrów.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, Windows.

Parametry:

- **laddr** - lokalny adres IPv4/6 lub podsieć CIDR;
- **lport** - numer portu lokalnego lub nazwa usługi;
- **raddr** - zdalny adres IPv4/6 lub podsieć CIDR;
- **rport** - numer portu zdalnego lub nazwa usługi;
- **state** - stan połączenia (*established*, *syn_sent*, *syn_recv*, *fin_wait1*, *fin_wait2*, *time_wait*, *close*, *close_wait*, *last_ack*, *listen*, *closing*).

Przykład:

```
net.tcp.socket.count[,80,,,established] #liczba połączeń z lokalnym portem TCP 80 w stanie established
```

net.udp.listen[port]

 Sprawdza, czy ten port UDP jest w stanie LISTEN.
 Zwracane wartości: 0 - nie jest w stanie LISTEN; 1 - jest w stanie LISTEN.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, MacOS X.

Parametr:

- **port** - numer portu UDP

Przykład:

```
net.udp.listen[68]
```

net.udp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i odpowiada na żądania UDP.
 Zwracane wartości: 0 - usługa jest niedostępna; 1 - usługa działa.
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie `127.0.0.1`);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
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>]

 Sprawdza wydajność usługi UDP.
 Zwracane wartości: *Float* (0 - usługa jest niedostępna; sekundy - liczba sekund spędzonych na oczekiwaniu na odpowiedź od usługi).
 Zobacz obsługiwane platformy.

Parametry:

- **service** - *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie `127.0.0.1`);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
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>]

 Zwraca liczbę gniazd UDP pasujących do parametrów.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, Windows.

Parametry:

- **laddr** - lokalny adres IPv4/6 lub podsieć CIDR;
- **lport** - numer portu lokalnego lub nazwa usługi;
- **raddr** - zdalny adres IPv4/6 lub podsieć CIDR;
- **rport** - numer portu zdalnego lub nazwa usługi;
- **state** - stan połączenia (*established*, *unconn*).

Przykład:

```
net.udp.socket.count[,,,,established] #zwraca liczbę gniazd UDP w stanie połączenia
```

proc.cpu.util[<name>,<user>,<type>,<cmdline>,<mode>,<zone>]

 Procentowe wykorzystanie CPU przez proces.
 Wartość zwracana: *Float*.
 Obsługiwane platformy: Linux, Solaris[6].

Parametry:

- **name** - nazwa procesu (domyślnie *all processes*);
- **user** - nazwa użytkownika (domyślnie *all users*);
- **type** - typ wykorzystania CPU: *total* (domyślnie), *user* lub *system*;
- **cmdline** - filtruj według wiersza poleceń (jest to wyrażenie regularne expression);
- **mode** - tryb zbierania danych: *avg1* (domyślnie), *avg5* lub *avg15*;
- **zone** - strefa docelowa: *current* (domyślnie) lub *all*. Ten parametr jest obsługiwany tylko w Solaris.

Uwagi:

- Zwracana wartość jest oparta na procentowym wykorzystaniu pojedynczego rdzenia CPU. Na przykład wykorzystanie CPU przez proces w pełni używający dwóch rdzeni wynosi 200%.
- Dane o wykorzystaniu CPU przez proces są zbierane przez collector, który obsługuje maksymalnie 1024 unikalne zapytania (według nazwy, użytkownika i wiersza poleceń). Zapytania, do których nie uzyskano dostępu w ciągu ostatnich 24 godzin, są usuwane z collectora.
- Ustawiając parametr zone na *current* (lub pozostawiając wartość domyślną), w przypadku gdy agent został skompilowany w Solaris bez obsługi stref, ale działa na nowszym Solaris, w którym strefy są obsługiwane, agent zwróci NOTSUPPORTED (agent nie może ograniczyć wyników wyłącznie do bieżącej strefy). Jednak *all* jest w tym przypadku obsługiwane.

Przykłady:

```
proc.cpu.util[,root] #wykorzystanie CPU przez wszystkie procesy uruchomione pod użytkownikiem "root"
proc.cpu.util[zabbix_server,zabbix] #wykorzystanie CPU przez wszystkie procesy zabbix_server uruchomione p
```

proc.get[<name>,<user>,<cmdline>,<mode>]

 Lista procesów systemu operacyjnego i ich parametrów. Może być używana do wykrywania niskiego poziomu.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Windows, OpenBSD, NetBSD.

Parametry:

- **name** - nazwa procesu (domyślnie *wszystkie procesy*).
- **user** - nazwa użytkownika (domyślnie *wszyscy użytkownicy*).
- **cmdline** - filtr według wiersza poleceń (jest to wyrażenie regularne expression). Ten parametr nie jest obsługiwany w systemie Windows; na innych platformach nie jest obsługiwany, jeśli tryb jest ustawiony na 'summary'.
- **mode** - możliwe wartości:
  - *process* (domyślnie);
  - *thread* (nieobsługiwane w NetBSD);

- *summary*. Zobacz listę parametrów procesów zwracanych dla każdego trybu i systemu operacyjnego.

Uwagi:

- Jeśli nie można pobrać wartości, na przykład z powodu błędu (proces już zakończył działanie, brak uprawnień, niepowodzenie wywołania systemowego), zostanie zwrócone −1.
- Zobacz uwagi dotyczące wybierania procesów za pomocą parametrów `name` i `cmdline` (dotyczy tylko Linux).

Przykłady:

```
proc.get[zabbix_server,zabbix,,process] #lista wszystkich procesów zabbix_server uruchomionych przez użytk
proc.get[java,,,thread] #lista wszystkich procesów Java, zwraca jeden wpis na wątek
proc.get[,zabbix,,summary] #dane zbiorcze dla procesów każdego typu uruchomionych przez użytkownika zabbix
```

proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]

 Pamięć używana przez proces w bajtach.
 Wartość zwracana: - *Integer* - przy *mode* ustawionym na *max*, *min*, *sum*; - *Float* - przy *mode* ustawionym na *avg*
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

Parametry:

- **name** - nazwa procesu (domyślnie *all processes*);
- **user** - nazwa użytkownika (domyślnie *all users*);
- **mode** - możliwe wartości: *avg*, *max*, *min* lub *sum* (domyślnie);
- **cmdline** - filtr według wiersza poleceń (jest to wyrażenie regularne expression);
- **memtype** - typ pamięci używany przez proces

Komentarze:

- Parametr `memtype` jest obsługiwany tylko w systemach Linux, FreeBSD, Solaris[6], AIX.
- Gdy kilka procesów korzysta ze współdzielonej pamięci, suma pamięci używanej przez procesy może dawać duże, nierealistyczne wartości.

Zobacz uwagi dotyczące wybierania procesów za pomocą parametrów `name` i `cmdline` (dotyczy tylko systemu Linux).

Gdy ta pozycja jest wywoływana z wiersza poleceń i zawiera parametr wiersza poleceń (np. przy użyciu trybu testowego agent: `zabbix_agentd -t proc.mem[,,,apache2]`), zostanie policzony jeden dodatkowy proces, ponieważ agent policzy sam siebie.

Przykłady:

```
proc.mem[,root] #pamięć używana przez wszystkie procesy uruchomione przez użytkownika "root"
proc.mem[zabbix_server,zabbix] #pamięć używana przez wszystkie procesy zabbix_server uruchomione przez uży
proc.mem[,oracle,max,oracleZABBIX] #pamięć używana przez proces zużywający najwięcej pamięci, uruchomiony
```

proc.num[<name>,<user>,<state>,<cmdline>,<zone>]

 Liczba procesów.
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris[6], HP-UX, AIX, OpenBSD, NetBSD, Windows.

Parametry:

- **name** - nazwa procesu (domyślnie *all processes*).
- **user** - nazwa użytkownika (domyślnie *all users*).
- **state** - możliwe wartości:
  - *all* (domyślnie);
  - *disk* - nieprzerywalny sen;
  - *run* - uruchomiony;
  - *sleep* - przerywalny sen;
  - *trace* - zatrzymany;
  - *zomb* - zombie.
- **cmdline** - filtruj według wiersza poleceń (jest to wyrażenie regularne expression).
- **zone** - strefa docelowa: *current* (domyślnie) lub *all*. Ten parametr jest obsługiwany tylko w Solaris.

Komentarze:

- Parametry stanu *disk* i *trace* są obsługiwane tylko w Linux, FreeBSD, OpenBSD, NetBSD.
- W Windows obsługiwane są tylko parametry `name` i `user`.
- Gdy ta pozycja jest wywoływana z wiersza poleceń i zawiera parametr wiersza poleceń (np. przy użyciu trybu testowego agenta: `zabbix_agentd -t proc.num[,,,apache2]`), zostanie policzony jeden dodatkowy proces, ponieważ agent policzy sam siebie.
- Ustawiając parametr `zone` na `current` (lub domyślnie) w przypadku, gdy agent został skompilowany w Solaris bez obsługi stref, ale działa na nowszym Solaris, w którym strefy są obsługiwane, agent zwróci NOTSUPPORTED (agent nie może ograniczyć wyników wyłącznie do bieżącej strefy). Jednak w tym przypadku obsługiwane jest *all*.

- Zobacz uwagi dotyczące wybierania procesów za pomocą parametrów `name` i `cmdline` (dotyczy tylko Linux).

Przykłady:

```
proc.num[,mysql] #liczba procesów uruchomionych przez użytkownika mysql
proc.num[apache2,www-data] #liczba procesów apache2 uruchomionych przez użytkownika www-data
proc.num[,oracle,sleep,oracleZABBIX] #liczba procesów w stanie sleep uruchomionych przez użytkownika Oracl
```

sensor[device,sensor,<mode>]

 Odczyt wartości czujnika sprzętowego.
 Zwracana wartość: *Float*.
 Obsługiwane platformy: Linux, OpenBSD.

Parametry:

- **device** - nazwa urządzenia;
- **sensor** - nazwa czujnika;
- **mode** - możliwe wartości: *avg*, *max* lub *min*.

Komentarze:

- Odczytuje `/proc/sys/dev/sensors` w Linux 2.4.
- Odczytuje `/sys/class/hwmon` w Linux 2.6+. Zobacz bardziej szczegółowy opis pozycji sensor w Linux.
- Odczytuje MIB `hw.sensors` w OpenBSD.
- W Linux 2.4, jeśli określisz mode, device i sensor są traktowane jako wyrażenia regularne.
- W Linux 2.6+ i OpenBSD, lub jeśli mode zostanie pominięty, device i sensor są traktowane dosłownie.

Przykłady:

```
sensor[w83781d-i2c-0-2d,temp1]
sensor[cpu0,temp0] #temperatura jednego procesora
sensor["cpu[0-2]$",temp,avg] #średnia temperatura pierwszych trzech procesorów
```

system.boottime

 Czas uruchomienia systemu.
 Zwracana wartość: *Liczba całkowita (znacznik czasu Unix)*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, MacOS X, OpenBSD, NetBSD.

system.cpu.discovery

 Lista wykrytych procesorów CPU/rdzeni CPU. Używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Zobacz obsługiwane platformy.

system.cpu.intr

 Przerwania urządzenia.
 Zwracana wartość: *Liczba całkowita*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.load[<cpu>,<mode>]

 Obciążenie CPU.
 Zwracana wartość: *Float*.
 Zobacz obsługiwane platformy.

Parametry:

- **cpu** — możliwe wartości: *all* (domyślnie) lub *percpu* (całkowite obciążenie podzielone przez liczbę aktywnych CPU);
- **mode** — możliwe wartości: *avg1* (średnia z jednej minuty, domyślnie), *avg5* lub *avg15*.

Przykład:

```
system.cpu.load[,avg5]
```

system.cpu.num[<type>]

 Liczba procesorów.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Parametr:

- **type** — możliwe wartości: *online* (domyślnie) lub *max*

Parametr typu *max* jest obsługiwany tylko w systemach Linux, FreeBSD, Solaris, MacOS X.

Przykład:

```
system.cpu.num
```

system.cpu.switches

 Liczba przełączeń kontekstu.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, NetBSD.

system.cpu.util[<cpu>,<type>,<mode>,<logical or physical>]

 Procent wykorzystania CPU.
 Zwracana wartość: *Float*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD, Windows.

Parametry:

- **cpu** - <*numer CPU*> lub *all* (domyślnie);
- **type** - możliwe wartości: *user* (domyślnie), *idle*, *nice*, *system*, *iowait*, *interrupt*, *softirq*, *steal*, *spin* (w OpenBSD), *guest* (w jądrach Linux 2.6.24 i nowszych) lub *guest_nice* (w jądrach Linux 2.6.33 i nowszych);
- **mode** - możliwe wartości: *avg1* (średnia jednominutowa, domyślnie), *avg5* lub *avg15*;
- **logical or physical** - możliwe wartości: *logical* (domyślnie) lub *physical*. Ten parametr jest obsługiwany tylko w AIX.

Uwagi:

- Parametr typu *nice* jest obsługiwany tylko w systemach Linux, FreeBSD, HP-UX, OpenBSD, NetBSD.
- Parametr typu *iowait* jest obsługiwany tylko w systemach Linux 2.6 i nowszych, Solaris, AIX.
- Parametr typu *interrupt* jest obsługiwany tylko w systemach Linux 2.6 i nowszych, FreeBSD, OpenBSD.
- Parametry typu *softirq*, *steal*, *guest*, *guest_nice* są obsługiwane tylko w systemach Linux 2.6 i nowszych.
- Parametry trybu *avg5* i *avg15* są obsługiwane w systemach Linux, FreeBSD, Solaris, HP-UX, AIX, OpenBSD, NetBSD.

Przykład:

```
system.cpu.util[0,user,avg5]
```

system.hostname[<type>,<transform>]

 Nazwa hosta systemu.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Parametry:

- **type** - możliwe wartości:
  - *netbios* (domyślnie w Windows);
  - *host* (domyślnie w Linux);
  - *shorthost* (zwraca część nazwy hosta przed pierwszą kropką, pełny ciąg dla nazw bez kropek);
  - *fqdn* (zwraca Fully Qualified Domain Name).
- **transform** - możliwe wartości:
  - *none* (domyślnie);
  - *lower* (konwertuje na małe litery).

Wartość jest pobierana przez odczyt `nodename` z wyjścia systemowego API uname().

Przykłady zwracanych wartości:

```
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>]

 Informacje o obudowie.
 Zwracana wartość: *String*.
 Obsługiwane platformy: Linux.

Parametr:

- **info** - możliwe wartości: *full* (domyślnie), *model*, *serial*, *type* lub *vendor*

Uwagi:

- Ten klucz pozycji zależy od dostępności tabeli SMBIOS.
- Nastąpi próba odczytu tabeli DMI z sysfs; jeśli dostęp do sysfs się nie powiedzie, zostanie podjęta próba bezpośredniego odczytu z pamięci.
- Wymagane są **uprawnienia root**, ponieważ wartość jest uzyskiwana przez odczyt z sysfs lub pamięci.

Przykład:

```
system.hw.chassis[full] → Hewlett-Packard HP Pro 3010 Small Form Factor PC CZXXXXXXXX Desktop
```

system.hw.cpu[<cpu>,<info>]

 Informacje o CPU.
 Zwracana wartość: *String* lub *Integer*.
 Obsługiwane platformy: Linux.

Parametry:

- **cpu** — <*numer CPU*> lub *all* (domyślnie);
- **info** — możliwe wartości: *full* (domyślnie), *curfreq*, *maxfreq*, *model* lub *vendor*.

Uwagi:

- Zbiera informacje z `/proc/cpuinfo` oraz `/sys/devices/system/cpu/[cpunum]/cpufreq/cpuinfo_max_freq`.
- Jeśli podano numer CPU oraz *curfreq* lub *maxfreq*, zwracana jest wartość liczbowa (Hz).

Przykład:

```
system.hw.cpu[0,vendor] → AuthenticAMD
```

system.hw.devices[<type>]

 Lista urządzeń PCI lub USB.
 Zwracana wartość: *Tekst*.
 Obsługiwane platformy: Linux.

Parametr:

- **type** - *pci* (domyślnie) lub *usb*

Zwraca dane wyjściowe narzędzia lspci lub lsusb (uruchamianego bez żadnych parametrów).

Przykład:

```
system.hw.devices → 00:00.0 Host bridge: Advanced Micro Devices [AMD] RS780 Host Bridge
```

system.hw.macaddr[<interface>,<format>]

 Lista adresów MAC.
 Wartość zwracana: *String*.
 Obsługiwane platformy: Linux.

Parametry:

- **interface** - *all* (domyślnie) lub wyrażenie regularne expression;
- **format** - *full* (domyślnie) lub *short*.

Komentarze:

- Wyświetla adresy MAC interfejsów, których nazwa pasuje do podanego wyrażenia regularnego `interface` (expression) (*all* wyświetla dla wszystkich interfejsów).
- Jeśli `format` ma wartość *short*, nazwy interfejsów i identyczne adresy MAC nie są wyświetlane.

Przykład:

```
system.hw.macaddr["eth0$",full] → [eth0] 00:11:22:33:44:55
```

system.localtime[<type>]

 Czas systemowy.
 Wartość zwracana: *Integer* - gdy `type` ma wartość *utc*; *String* - gdy `type` ma wartość *local*.
 Zobacz obsługiwane platformy.

Parametry:

- **type** - możliwe wartości:
  - *utc* - (domyślnie) czas od Epoki (00:00:00 UTC, 1 stycznia 1970), mierzony w sekundach;
  - *local* - czas w formacie 'yyyy-mm-dd,hh:mm:ss.nnn,+hh:mm'.

Musi być używane jako bierne sprawdzenie dla Zabbix agent; w Zabbix agent 2 może być skonfigurowane jako aktywne sprawdzenie.

Przykład:

```
system.localtime[local] #utwórz pozycję używając tego klucza, a następnie użyj jej do wyświetlenia czasu h
```

system.run[command,<mode>]

 Uruchamia określone polecenie na hoście.
 Wartość zwracana: wynik *Text* polecenia lub 1 - przy `mode` ustawionym na *nowait* (niezależnie od wyniku polecenia).
 Zobacz obsługiwane platformy.

Parametry:

- **command** - polecenie do wykonania;
- **mode** - możliwe wartości:

- *wait* - czekaj na zakończenie wykonania (domyślnie);
- *nowait* - nie czekaj.

Uwagi:

- Ta pozycja jest domyślnie wyłączona. Dowiedz się, jak włączyć je.
- Wartość zwracana przez pozycję to standardowe wyjście wraz ze standardowym błędem wygenerowanym przez polecenie. Sprawdzanie kodu zakończenia nie jest wykonywane.
- Aby przetwarzanie przebiegało poprawnie, wartość zwracana przez polecenie musi mieć typ danych `text`. Pusty wynik jest również dozwolony.
- Wartość zwracana jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.
- Zobacz także: Wykonywanie poleceń.

Przykład:

```
system.run[ls -l /] #zwraca szczegółową listę plików katalogu głównego
```

system.stat[resource,<type>]

 Statystyki systemowe.
 Wartość zwracana: *Integer* lub *float*.
 Obsługiwane platformy: AIX.

Parametry:

- **ent** - liczba jednostek procesora, do których ta partycja ma prawo (float).
- **kthr,<type>** - informacje o stanach wątków jądra:
  - *r* - średnia liczba wątków jądra gotowych do uruchomienia (float);
  - *b* - średnia liczba wątków jądra umieszczonych w kolejce oczekiwania Virtual Memory Manager (float).
- **memory,<type>** - informacje o wykorzystaniu pamięci wirtualnej i fizycznej:
  - *avm* - aktywne strony wirtualne (integer);
  - *fre* - rozmiar listy wolnych zasobów (integer).
- **page,<type>** - informacje o błędach stron i aktywności stronicowania:
  - *fi* - liczba wejść stron z pliku na sekundę (float);
  - *fo* - liczba wyjść stron do pliku na sekundę (float);
  - *pi* - strony wczytane z przestrzeni stronicowania (float);
  - *po* - strony zapisane do przestrzeni stronicowania (float);
  - *fr* - zwolnione strony (zastępowanie stron) (float);
  - *sr* - strony skanowane przez algorytm zastępowania stron (float).
- **faults,<type>** - częstotliwość pułapek i przerwań:
  - *in* - przerwania urządzeń (float);
  - *sy* - wywołania systemowe (float);
  - *cs* - przełączenia kontekstu wątków jądra (float).
- **cpu,<type>** - podział procentowego wykorzystania czasu procesora:
  - *us* - czas użytkownika (float);
  - *sy* - czas systemowy (float);
  - *id* - czas bezczynności (float);
  - *wa* - czas bezczynności, w którym system miał oczekujące żądania I/O dysku/NFS (float);
  - *pc* - liczba wykorzystanych fizycznych procesorów (float);
  - *ec* - procent wykorzystanej przydzielonej pojemności (float);
  - *lbusy* - wskazuje procent wykorzystania logicznego procesora lub procesorów, który wystąpił podczas wykonywania na poziomie użytkownika i systemu (float);
  - *app* - wskazuje dostępne fizyczne procesory w puli współdzielonej (float).
- **disk,<type>** - statystyki dysku:
  - *bps* - wskazuje ilość danych przesłanych (odczytanych lub zapisanych) do dysku w bajtach na sekundę (integer);
  - *tps* - wskazuje liczbę transferów na sekundę wysłanych do fizycznego dysku/taśmy (float).

Uwagi:

- Zwróć uwagę na następujące ograniczenia w tych pozycjach:
  - `system.stat[cpu,app]` - obsługiwane tylko na AIX LPAR typu ”Shared”;
  - `system.stat[cpu,ec]` - obsługiwane na AIX LPAR typu ”Shared” i ”Dedicated” (”Dedicated” zawsze zwraca 100 (procent));
  - `system.stat[cpu,lbusy]` - obsługiwane tylko na AIX LPAR typu ”Shared”;
  - `system.stat[cpu,pc]` - obsługiwane na AIX LPAR typu ”Shared” i ”Dedicated”;
  - `system.stat[ent]` - obsługiwane na AIX LPAR typu ”Shared” i ”Dedicated”.

system.sw.arch

 Informacja o architekturze oprogramowania.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Informacja jest pobierana za pomocą funkcji `uname()`.

Przykład:

```
system.sw.arch → i686
```

system.sw.os[<info>]

 Informacje o systemie operacyjnym.
 Wartość zwracana: *String*.
 Obsługiwane platformy: Linux, Windows.

Parametr:

- **info** - możliwe wartości: *full* (domyślnie), *short* lub *name*

Informacja info jest pobierana z (należy pamiętać, że nie wszystkie pliki i opcje są obecne we wszystkich dystrybucjach):

- `/proc/version` (*full*) w systemie Linux;
- `/proc/version_signature` (*short*) w systemie Linux;
- parametr PRETTY_NAME z `/etc/os-release` w systemach Linux obsługujących ten plik lub `/etc/issue.net` (*name*);
- klucz rejestru `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion` w systemie Windows.

Przykłady:

```
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

 Szczegółowe informacje o systemie operacyjnym (wersja, typ, nazwa dystrybucji, wersja główna i pomocnicza itp.).
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, Windows.

system.sw.packages[<regexp>,<manager>,<format>]

 Lista zainstalowanych pakietów.
 Wartość zwracana: *Text*.
 Obsługiwane platformy: Linux.

Parametry:

- **regexp** - *all* (domyślnie) lub wyrażenie regularne expression;
- **manager** - *all* (domyślnie) lub menedżer pakietów;
- **format** - *full* (domyślnie) lub *short*.

Uwagi:

- Wyświetla (w kolejności alfabetycznej) zainstalowane pakiety, których nazwa pasuje do podanego wyrażenia regularnego expression (*all* wyświetla wszystkie).
- Obsługiwane menedżery pakietów (wykonywane polecenie):
  - pkg (`dpkg --get-selections`);
  - pkgtool (`ls /var/log/packages`);
  - rpm (`rpm -qa`);
  - pacman (`pacman -Q`);
  - portage.
- Jeśli `format` ma wartość *full*, pakiety są grupowane według menedżerów pakietów (każdy menedżer w osobnym wierszu, rozpoczynającym się od jego nazwy w nawiasach kwadratowych).
- Jeśli `format` ma wartość *short*, pakiety nie są grupowane i są wyświetlane w jednym wierszu.

Przykład:

```
system.sw.packages[mini,dpkg,short] → python-minimal, python2.6-minimal, ubuntu-minimal
```

system.sw.packages.get[<regexp>,<manager>]

 Szczegółowa lista zainstalowanych pakietów.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux.

Parametry:

- **regexp** - *all* (domyślnie) lub wyrażenie regularne expression;
- **manager** - *all* (domyślnie) lub menedżer pakietów (możliwe wartości: *rpm*, *dpkg*, *pkgtool*, *pacman* lub *portage*).

Komentarze:

- Zwraca niesformatowany JSON z zainstalowanymi pakietami, których nazwa pasuje do podanego wyrażenia regularnego.
- Wynik jest tablicą obiektów, z których każdy zawiera następujące klucze: name, manager, version, size, architecture, build-time i installtime (zobacz więcej szczegółów).

system.swap.in[<device>,<type>]

 Statystyki swap-in (z urządzenia do pamięci).
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, OpenBSD.

Parametry:

- **device** - określa urządzenie używane do swapowania (tylko Linux) lub *all* (domyślnie);
- **type** - możliwe wartości:
  - *count* (liczba swap-inów, domyślnie na platformach innych niż Linux);
  - *sectors* (sektory wczytane do swap);
  - *pages* (strony wczytane do swap, domyślnie na Linux).

Komentarze:

- Źródłem tych informacji są:
  - `/proc/swaps`, `/proc/partitions`, `/proc/stat` (Linux 2.4);
  - `/proc/swaps`, `/proc/diskstats`, `/proc/vmstat` (Linux 2.6).
- Należy pamiętać, że *pages* będzie działać tylko wtedy, gdy nie określono urządzenia.
- Parametr typu *sectors* jest obsługiwany tylko na Linux.

Przykład:

```
system.swap.in[,pages]
```

system.swap.out[<device>,<type>]

 Statystyki swap-out (z pamięci na urządzenie).
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, OpenBSD.

Parametry:

- **device** - określa urządzenie używane do swapowania (tylko Linux) lub *all* (domyślnie);
- **type** - możliwe wartości:
  - *count* (liczba operacji swapout, domyślnie na platformach innych niż Linux);
  - *sectors* (sektory zapisane do swap);
  - *pages* (strony zapisane do swap, domyślnie w systemie Linux).

Uwagi:

- Źródłem tych informacji są:
  - `/proc/swaps`, `/proc/partitions`, `/proc/stat` (Linux 2.4);
  - `/proc/swaps`, `/proc/diskstats`, `/proc/vmstat` (Linux 2.6).
- Należy pamiętać, że *pages* będzie działać tylko wtedy, gdy nie określono urządzenia.
- Parametr typu *sectors* jest obsługiwany tylko w systemie Linux.

Przykład:

```
system.swap.out[,pages]
```

system.swap.size[<device>,<type>]

 Rozmiar przestrzeni wymiany w bajtach lub w procentach całości.
 Wartość zwracana: *Integer* - dla bajtów; *Float* - dla procentów.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD, Windows.

Parametry:

- **device** - określa urządzenie używane do swapowania (tylko FreeBSD) lub *all* (domyślnie);
- **type** - możliwe wartości:
  - *free* (wolna przestrzeń swap, domyślnie);
  - *pfree* (wolna przestrzeń swap, w procentach);
  - *pused* (użyta przestrzeń swap, w procentach);
  - *total* (całkowita przestrzeń swap);
  - *used* (użyta przestrzeń swap).

Uwagi:

- Należy pamiętać, że *pfree*, *pused* nie są obsługiwane w systemie Windows, jeśli rozmiar swap wynosi 0.
- Jeśli device nie zostanie określone, agent Zabbix będzie uwzględniał tylko urządzenia swap (pliki), a pamięć fizyczna zostanie zignorowana. Na przykład w systemach Solaris polecenie `swap -s` obejmuje część pamięci fizycznej i urządzenia swap (w przeciwieństwie do `swap -l`).

Przykład:

```
system.swap.size[,pfree] → procent wolnej przestrzeni swap
```

system.uname

 Identyfikacja systemu.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Uwagi:

- W systemach UNIX wartość tej pozycji jest uzyskiwana za pomocą wywołania systemowego uname().
- W systemie Windows pozycja zwraca architekturę systemu operacyjnego, natomiast w systemach UNIX zwraca architekturę CPU.

Przykłady:

```
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

 Czas działania systemu w sekundach.
 Zwracana wartość: *Integer*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, MacOS X, OpenBSD, NetBSD, Windows.

W konfiguracji pozycji użyj jednostek **s** lub **uptime**, aby uzyskać czytelne wartości.

system.users.num

 Liczba zalogowanych użytkowników.
 Wartość zwracana: *Integer*.
 Zobacz obsługiwane platformy.

Do uzyskania tej wartości po stronie agent używane jest polecenie **who**.

vfs.dev.discovery

 Lista urządzeń blokowych i ich typów. Używane do wykrywania niskiego poziomu.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux.

vfs.dev.get[<mode>,<regexp>]

 Lista urządzeń blokowych ze szczegółowymi informacjami. Używane do niskopoziomowego wykrywania.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux, AIX.

Parametry:

- **mode** - możliwe wartości:
  - *disks* (domyślnie) - zwraca listę dysków;
  - *disk_stats* - zwraca statystyki dla każdego dysku;
  - *devices* (tylko Linux) - zwraca listę urządzeń blokowych (dysków i partycji);
  - *device_stats* (tylko Linux) - zwraca statystyki dla każdego urządzenia (dysków i partycji).
- **regexp** - wyrażenie regularne expression używane do filtrowania urządzeń blokowych według ich nazwy.

vfs.dev.read[<device>,<type>,<mode>]

 Statystyki odczytu dysku.
 Wartość zwracana: - *Integer* - dla `type` ustawionego na *sectors*, *operations*, *bytes*; - *Float* - dla `type` ustawionego na *sps*, *ops*, *bps*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Parametry:

- **device** - urządzenie dyskowe (domyślnie *all* [3]);
- **type** - możliwe wartości: *sectors*, *operations*, *bytes*, *sps*, *ops* lub *bps* (*sps*, *ops*, *bps* oznaczają odpowiednio: sektory, operacje, bajty na sekundę);
- **mode** - możliwe wartości: *avg1* (średnia z jednej minuty, domyślnie), *avg5* lub *avg15*. Ten parametr jest obsługiwany tylko wtedy, gdy `type` ma wartość: sps, ops, bps.

Uwagi:

- Jeśli używany jest interwał aktualizacji wynoszący trzy godziny lub więcej[2], ta pozycja zawsze zwróci 0.
- Parametry typu *sectors* i *sps* są obsługiwane tylko w systemie Linux.
- Parametr typu *ops* jest obsługiwany tylko w systemach Linux i FreeBSD.
- Parametr typu *bps* jest obsługiwany tylko w systemie FreeBSD.
- Parametr typu *bytes* jest obsługiwany tylko w systemach FreeBSD, Solaris, AIX, OpenBSD.
- Parametr `mode` jest obsługiwany tylko w systemach Linux, FreeBSD.
- Można używać nazw względnych urządzeń, na przykład `sda`, a także opcjonalnego prefiksu `/dev/` (na przykład `/dev/sda`).
- Obsługiwane są logiczne woluminy LVM.
- Domyślne wartości parametru 'type' dla różnych systemów operacyjnych:
  - *AIX* - operations;

    - *FreeBSD* - bps;
    - *Linux* - sps;
    - *OpenBSD* - operations;
    - *Solaris* - bytes.
- *sps*, *ops* i *bps* na obsługiwanych platformach są ograniczone do 1024 urządzeń (1023 indywidualnych i jednego dla *all*).

Przykład:

```
vfs.dev.read[,operations]
```

vfs.dev.write[<device>,<type>,<mode>]

 Statystyki zapisu na dysku.
 Wartość zwracana: *Integer* - dla `type` ustawionego na *sectors*, *operations*, *bytes*; *Float* - dla `type` ustawionego na *sps*, *ops*, *bps*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, AIX, OpenBSD.

Parametry:

- **device** - urządzenie dyskowe (domyślnie *all* [3]);
- **type** - możliwe wartości: *sectors*, *operations*, *bytes*, *sps*, *ops* lub *bps* (*sps*, *ops*, *bps* oznaczają odpowiednio: sektory, operacje, bajty na sekundę);
- **mode** - możliwe wartości: *avg1* (średnia z jednej minuty, domyślnie), *avg5* lub *avg15*. Ten parametr jest obsługiwany tylko dla `type` ustawionego na: sps, ops, bps.

Uwagi:

- Jeśli używany jest interwał aktualizacji wynoszący trzy godziny lub więcej[2], ta pozycja zawsze zwróci 0.
- Parametry typu *sectors* i *sps* są obsługiwane tylko w systemie Linux.
- Parametr typu *ops* jest obsługiwany tylko w systemach Linux i FreeBSD.
- Parametr typu *bps* jest obsługiwany tylko w systemie FreeBSD.
- Parametr typu *bytes* jest obsługiwany tylko w systemach FreeBSD, Solaris, AIX, OpenBSD.
- Parametr `mode` jest obsługiwany tylko w systemach Linux, FreeBSD.
- Można używać nazw względnych urządzeń (na przykład `sda`) oraz opcjonalnego prefiksu `/dev/` (na przykład `/dev/sda`).
- Obsługiwane są logiczne woluminy LVM.
- Domyślne wartości parametru `type` dla różnych systemów operacyjnych:
    - *AIX* - operations
    - *FreeBSD* - bps
    - *Linux* - sps
    - *OpenBSD* - operations
    - *Solaris* - bytes
- *sps*, *ops* i *bps* na obsługiwanych platformach są ograniczone do 1024 urządzeń (1023 indywidualnych i jednego dla *all*).

Przykład:

```
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>]

 Liczba wpisów w katalogu.
 Zwracana wartość: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **dir** - bezwzględna ścieżka do katalogu;
- **regex incl** - wyrażenie regularne expression opisujące wzorzec nazwy jednostki (pliku, katalogu, dowiązania symbolicznego), która ma zostać uwzględniona; jeśli puste, uwzględniane są wszystkie jednostki (wartość domyślna);
- **regex excl** - wyrażenie regularne expression opisujące wzorzec nazwy jednostki (pliku, katalogu, dowiązania symbolicznego), która ma zostać wykluczona; jeśli puste, żadna jednostka nie jest wykluczana (wartość domyślna);
- **types incl** - typy wpisów katalogu, które mają być zliczane; możliwe wartości:
    - *file* - zwykły plik;
    - *dir* - podkatalog;
    - *sym* - dowiązanie symboliczne;
    - *sock* - gniazdo;
    - *bdev* - urządzenie blokowe;
    - *cdev* - urządzenie znakowe;
    - *fifo* - FIFO;
    - *dev* - synonim „bdev,cdev”;
    - *all* - wszystkie typy (domyślnie), tj. „file,dir,sym,sock,bdev,cdev,fifo”. Wiele typów należy oddzielić przecinkiem i ująć w cudzysłów.

- **types excl** - typy wpisów katalogu (zobacz `types incl`), których NIE należy zliczać. Jeśli dany typ wpisu znajduje się zarówno w `types incl`, jak i w `types excl`, wpisy katalogu tego typu NIE są zliczane.
- **max depth** - maksymalna głębokość podkatalogów, które mają być przeszukiwane:
  - **-1** (domyślnie) - bez ograniczeń;
  - **0** - bez przechodzenia do podkatalogów.
- **min size** - minimalny rozmiar (w bajtach) pliku, który ma zostać zliczony. Mniejsze pliki nie będą zliczane. Można używać sufiksów pamięci.
- **max size** - maksymalny rozmiar (w bajtach) pliku, który ma zostać zliczony. Większe pliki nie będą zliczane. Można używać sufiksów pamięci.
- **min age** - minimalny wiek (w sekundach) wpisu katalogu, który ma zostać zliczony. Nowsze wpisy nie będą zliczane. Można używać sufiksów czasu.
- **max age** - maksymalny wiek (w sekundach) wpisu katalogu, który ma zostać zliczony. Wpisy w tym wieku i starsze nie będą zliczane (czas modyfikacji). Można używać sufiksów czasu.
- **regex excl dir** - wyrażenie regularne expression opisujące wzorzec nazwy katalogu, który ma zostać wykluczony. Cała zawartość katalogu zostanie wykluczona (w przeciwieństwie do `regex_excl`).

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, które mogą stanowić zagrożenie bezpieczeństwa.
- Zmienne środowiskowe, np. `%APP_HOME%`, `$HOME` i `%TEMP%`, nie są obsługiwane.
- Pseudokatalogi „.” i „..” nigdy nie są zliczane.
- Podczas przechodzenia przez katalogi dowiązania symboliczne nigdy nie są śledzone.
- Zarówno `regex incl`, jak i `regex excl` są stosowane do plików i katalogów podczas obliczania liczby wpisów, ale są ignorowane przy wyborze podkatalogów do przeszukania (jeśli `regex incl` ma wartość `(?i)^.+\.zip$`, a `max depth` nie jest ustawione, przeszukiwane będą wszystkie podkatalogi, ale zliczane będą tylko pliki typu zip).
- Czas wykonania jest ograniczony wartością limitu czasu w konfiguracji agent (3 sekundy). Ponieważ przeszukiwanie dużego katalogu może trwać dłużej, dane nie zostaną zwrócone, a pozycja stanie się nieobsługiwana. Częściowa liczba nie zostanie zwrócona.
- Podczas filtrowania według rozmiaru tylko zwykłe pliki mają znaczące rozmiary. W systemach Linux i BSD katalogi również mają rozmiary różne od zera (zwykle kilka KB). Urządzenia mają rozmiar równy zero, np. rozmiar `/dev/sda1` nie odzwierciedla rozmiaru odpowiedniej partycji. Dlatego podczas używania `<min_size>` i `<max_size>` zaleca się określenie `<types_incl>` jako *„file”*, aby uniknąć niespodziewanych wyników.

Przykłady:

```
vfs.dir.count[/dev] #monitoruje liczbę urządzeń w /dev (Linux)
vfs.dir.count["C:\Users\ADMINI~1\AppData\Local\Temp"] #monitoruje liczbę plików w katalogu tymczasowym
```

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>]

 Lista wpisów katalogu.
 Zwracana wartość: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametry:

- **dir** - bezwzględna ścieżka do katalogu;
- **regex incl** - wyrażenie regularne expression opisujące wzorzec nazwy jednostki (pliku, katalogu, dowiązania symbolicznego), która ma zostać uwzględniona; jeśli jest puste, uwzględniane są wszystkie jednostki (wartość domyślna);
- **regex excl** - wyrażenie regularne expression opisujące wzorzec nazwy jednostki (pliku, katalogu, dowiązania symbolicznego), która ma zostać wykluczona; jeśli jest puste, żadna jednostka nie jest wykluczana (wartość domyślna);
- **types incl** - typy wpisów katalogu, które mają zostać wyświetlone; możliwe wartości:
  - *file* - zwykły plik;
  - *dir* - podkatalog;
  - *sym* - dowiązanie symboliczne;
  - *sock* - gniazdo;
  - *bdev* - urządzenie blokowe;
  - *cdev* - urządzenie znakowe;
  - *fifo* - FIFO;
  - *dev* - synonim wartości „bdev,cdev”;
  - *all* - wszystkie typy (domyślnie), tj. „file,dir,sym,sock,bdev,cdev,fifo”. Wiele typów należy oddzielić przecinkami i ująć w cudzysłów.
- **types excl** - typy wpisów katalogu (zobacz `types incl`), których NIE należy wyświetlać. Jeśli dany typ wpisu znajduje się zarówno w `types incl`, jak i w `types excl`, wpisy katalogu tego typu NIE zostaną wyświetlone.
- **max depth** - maksymalna głębokość podkatalogów, które mają zostać przeszukane:
  - **-1** (domyślnie) - bez ograniczeń;
  - **0** - bez schodzenia do podkatalogów.

- **min size** - minimalny rozmiar (w bajtach) pliku, który ma zostać wyświetlony. Mniejsze pliki nie zostaną wyświetlone. Można używać sufiksów pamięci.
- **max size** - maksymalny rozmiar (w bajtach) pliku, który ma zostać wyświetlony. Większe pliki nie zostaną wyświetlone. Można używać sufiksów pamięci.
- **min age** - minimalny wiek (w sekundach) wpisu katalogu, który ma zostać wyświetlony. Nowsze wpisy nie zostaną wyświetlone. Można używać sufiksów czasu.
- **max age** - maksymalny wiek (w sekundach) wpisu katalogu, który ma zostać wyświetlony. Wpisy w tym wieku i starsze nie zostaną wyświetlone (czas modyfikacji). Można używać sufiksów czasu.
- **regex excl dir** - wyrażenie regularne expression opisujące wzorzec nazwy katalogu, który ma zostać wykluczony. Cała zawartość katalogu zostanie wykluczona (w przeciwieństwie do `regex excl`).

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Zmienne środowiskowe, takie jak `%APP_HOME%`, `$HOME` i `%TEMP%`, nie są obsługiwane.
- Pseudokatalogi „.” i „..” nigdy nie są wyświetlane.
- Dowiązania symboliczne nigdy nie są używane podczas przeszukiwania katalogów.
- Zarówno `regex incl`, jak i `regex excl` są stosowane do plików i katalogów podczas generowania listy wpisów, ale są ignorowane przy wyborze podkatalogów do przeszukania (jeśli `regex incl` ma wartość `(?i)^.+\.zip$`, a `max depth` nie jest ustawione, przeszukiwane będą wszystkie podkatalogi, ale zliczane będą tylko pliki typu zip).
- Czas wykonania jest ograniczony wartością limitu czasu w konfiguracji agent. Ponieważ przeszukiwanie dużego katalogu może trwać dłużej, dane nie zostaną zwrócone, a pozycja zmieni się na nieobsługiwaną. Lista częściowa nie zostanie zwrócona.
- Podczas filtrowania według rozmiaru tylko zwykłe pliki mają znaczące rozmiary. W systemach Linux i BSD katalogi również mają rozmiary różne od zera (zwykle kilka KB). Urządzenia mają rozmiar równy zero, np. rozmiar `/dev/sda1` nie odzwierciedla rozmiaru odpowiedniej partycji. Dlatego podczas używania `min size` i `max size` zaleca się określenie `types incl` jako „*file*”, aby uniknąć niespodzianek.

Przykłady:

```
vfs.dir.get[/dev] #pobiera listę urządzeń w /dev (Linux)
vfs.dir.get["C:\Users\ADMINI~1\AppData\Local\Temp"] #pobiera listę plików w katalogu tymczasowym
```

vfs.dir.size[dir,<regex incl>,<regex excl>,<mode>,<max depth>,<regex excl dir>]

 Rozmiar katalogu (w bajtach).
 Wartość zwracana: *Integer*.
 Obsługiwane platformy: Linux, Windows. Pozycja może działać także na innych platformach typu UNIX.

Parametry:

- **dir** - bezwzględna ścieżka do katalogu;
- **regex incl** - wyrażenie regularne expression opisujące wzorzec nazwy encji (pliku, katalogu, dowiązania symbolicznego) do uwzględnienia; jeśli puste, uwzględnij wszystkie (wartość domyślna);
- **regex excl** - wyrażenie regularne expression opisujące wzorzec nazwy encji (pliku, katalogu, dowiązania symbolicznego) do wykluczenia; jeśli puste, nie wykluczaj żadnej (wartość domyślna);
- **mode** - możliwe wartości:
  - *apparent* (domyślnie) - zwraca rozmiary plików widoczne, a nie zajętość dysku (działa jak `du -sb dir`);
  - *disk* - zwraca zajętość dysku (działa jak `du -s -B1 dir`). W przeciwieństwie do polecenia `du`, pozycja vfs.dir.size uwzględnia pliki ukryte podczas obliczania rozmiaru katalogu (działa jak `du -sb .[^.]* *` w obrębie dir).
- **max depth** - maksymalna głębokość przeszukiwania podkatalogów:
  - **-1** (domyślnie) - bez ograniczeń;
  - **0** - bez schodzenia do podkatalogów.
- **regex excl dir** - wyrażenie regularne expression opisujące wzorzec nazwy katalogu do wykluczenia. Cała zawartość katalogu zostanie wykluczona (w przeciwieństwie do `regex excl`)

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Obliczane są tylko katalogi, dla których użytkownik *zabbix* ma co najmniej uprawnienie do odczytu. Dla katalogów z samym uprawnieniem do odczytu obliczany jest rozmiar samego katalogu. Katalogi z uprawnieniami do odczytu i wykonywania są obliczane wraz z zawartością.
- W przypadku dużych katalogów lub wolnych dysków ta pozycja może przekroczyć limit czasu z powodu ustawienia Timeout w plikach konfiguracyjnych agent i serwer/proxy. W razie potrzeby zwiększ wartości limitu czasu.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykłady:

```
vfs.dir.size[/tmp,log] #oblicza rozmiar wszystkich plików w /tmp zawierających w nazwie 'log'
vfs.dir.size[/tmp,log,^.+\.old$] #oblicza rozmiar wszystkich plików w /tmp zawierających w nazwie 'log', z
```

vfs.file.cksum[file,<mode>]

 Suma kontrolna pliku, obliczana algorytmem UNIX cksum.
 Wartość zwracana: *Integer* - gdy `mode` ma wartość *crc32*, *String* - gdy `mode` ma wartość *md5*, *sha256*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **mode** - *crc32* (domyślnie), *md5* lub *sha256*.

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.cksum[/etc/passwd]
```

Przykłady zwracanych wartości (odpowiednio crc32/md5/sha256):

```
675436101
9845acf68b73991eb7fd7ee0ded23c44
ae67546e4aac995e5c921042d0cf0f1f7147703aa42bfbfb65404b30f238f2dc
```

vfs.file.contents[file,<encoding>]

 Pobieranie zawartości pliku[7].
 Wartość zwracana: *Text*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **encoding** - identyfikator strony kodowej identyfikator.

Komentarze:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Wartość zwracana jest ograniczona do 16 MB (w tym końcowe białe znaki, które są obcinane); obowiązują również limity bazy danych.
- Zwracany jest pusty ciąg, jeśli plik jest pusty lub zawiera wyłącznie znaki LF/CR.
- Znacznik kolejności bajtów (BOM) jest wykluczony z wyniku.

Przykład:

```
vfs.file.contents[/etc/passwd]
```

vfs.file.exists[file,<types incl>,<types excl>]

 Sprawdza, czy plik istnieje.
 Wartość zwracana: 0 - nie znaleziono; 1 - istnieje plik określonego typu.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **types incl** - lista typów plików do uwzględnienia, możliwe wartości:
  - *file* (zwykły plik, domyślnie (jeśli `types_excl` nie jest ustawione));
  - *dir* (katalog);
  - *sym* (dowiązanie symboliczne);
  - *sock* (gniazdo);
  - *bdev* (urządzenie blokowe);
  - *cdev* (urządzenie znakowe);
  - *fifo* (FIFO);
  - *dev* (synonim "bdev,cdev");
  - *all* (wszystkie wymienione typy, domyślnie, jeśli `types_excl` jest ustawione).
- **types excl** - lista typów plików do wykluczenia, zobacz `types_incl` dla możliwych wartości (domyślnie żadne typy nie są wykluczane)

Uwagi:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Wiele typów musi być oddzielonych przecinkiem, a cały zestaw ujęty w cudzysłów "".
- Jeśli ten sam typ znajduje się zarówno w <types_incl>, jak i <types_excl>, pliki tego typu są wykluczane.

- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykłady:

```
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]

 Zwraca informacje o pliku.
 Wartość zwracana: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametr:

- **file** - pełna ścieżka do pliku

Komentarze:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Obsługiwane typy plików w systemach podobnych do UNIX: zwykły plik, katalog, dowiązanie symboliczne, gniazdo, urządzenie blokowe, urządzenie znakowe, FIFO.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.get[/etc/passwd] #zwraca JSON z informacjami o pliku /etc/passwd (typ, użytkownik, uprawnienia, S
```

vfs.file.md5sum[file]

 Suma kontrolna MD5 pliku.
 Zwracana wartość: Ciąg znaków (hash MD5 pliku).
 Zobacz obsługiwane platformy.

Parametr:

- **file** - pełna ścieżka do pliku

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.md5sum[/usr/local/etc/zabbix_agentd.conf]
```

Przykład zwracanej wartości:

```
b5052decb577e0fffd622d6ddc017e82
```

vfs.file.owner[file,<ownertype>,<resulttype>]

 Pobiera właściciela pliku.
 Wartość zwracana: *String*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **ownertype** - *user* (domyślnie) lub *group* (tylko Unix);
- **resulttype** - *name* (domyślnie) lub *id*; dla id - zwraca uid/gid w Unix, SID w Windows.

Komentarze:

- W systemie Windows ta pozycja podąża za ścieżkami UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykłady:

```
vfs.file.owner[/tmp/zabbix_server.log] #zwraca właściciela pliku /tmp/zabbix_server.log
vfs.file.owner[/tmp/zabbix_server.log,,id] #zwraca identyfikator właściciela pliku /tmp/zabbix_server.log
```

vfs.file.permissions[file]

 Zwraca 4-cyfrowy ciąg zawierający liczbę ósemkową z uprawnieniami UNIX.
 Zwracana wartość: *String*.
 Obsługiwane platformy: Linux. Pozycja może działać na innych platformach typu UNIX.

Parametry:

- **file** — pełna ścieżka do pliku

Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.permissions[/etc/passwd] #zwraca uprawnienia dla /etc/passwd, na przykład '0644'
```

vfs.file.regexp[file,regexp,<encoding>,<start line>,<end line>,<output>]

 Pobiera ciąg znaków z pliku[7].
 Wartość zwracana: Wiersz zawierający dopasowany ciąg znaków albo wartość określona przez opcjonalny parametr `output`.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej identifier;
- **start line** - numer pierwszego wiersza do przeszukania (domyślnie pierwszy wiersz pliku);
- **end line** - numer ostatniego wiersza do przeszukania (domyślnie ostatni wiersz pliku);
- **output** - opcjonalny szablon formatowania wyniku. Sekwencja ucieczki **\0** jest zastępowana dopasowaną częścią tekstu (od pierwszego znaku, w którym zaczyna się dopasowanie, do znaku, w którym się kończy), natomiast sekwencja ucieczki **\N** (gdzie N=1...9) jest zastępowana N-tym dopasowanym □□□□?

vfs.file.regmatch[file,regexp,<encoding>,<start line>,<end line>]

 Znajduje ciąg znaków w pliku[7].
 Zwracane wartości: 0 - nie znaleziono dopasowania; 1 - znaleziono.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec;
- **encoding** - identyfikator strony kodowej identifier;
- **start line** - numer pierwszej linii do przeszukania (domyślnie pierwsza linia pliku);
- **end line** - numer ostatniej linii do przeszukania (domyślnie ostatnia linia pliku).

Uwagi:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.
- Znacznik kolejności bajtów (BOM) jest ignorowany.

Przykład:

```
vfs.file.regmatch[/var/log/app.log,error]
```

vfs.file.size[file,<mode>]

 Rozmiar pliku (w bajtach).
 Zwracana wartość: *Integer*.
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **mode** - możliwe wartości:
  - *bytes* (domyślnie);
  - *lines* (puste wiersze są również liczone).

Komentarze:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Plik musi mieć uprawnienia do odczytu dla użytkownika *zabbix*.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.size[/var/log/syslog]
```

vfs.file.time[file,<mode>]

 Informacje o czasie pliku.
 Wartość zwracana: *Integer* (znacznik czasu Unix).
 Zobacz obsługiwane platformy.

Parametry:

- **file** - pełna ścieżka do pliku;
- **mode** - możliwe wartości:
  - *modify* (domyślnie) - czas ostatniej modyfikacji zawartości pliku (domyślnie);

  - *access* - czas ostatniego odczytu pliku;
  - *change* - czas ostatniej zmiany właściwości pliku.

Komentarze:

- W systemie Windows ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa.
- Limit rozmiaru pliku zależy od obsługi dużych plików.

Przykład:

```
vfs.file.time[/etc/passwd,modify]
```

vfs.fs.discovery

 Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania. Używana do wykrywania niskiego poziomu.
 Zwracana wartość: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

vfs.fs.get[<mode>,<mountpoint>]

 Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania, a w trybie *full* także z informacjami o dostępnej przestrzeni dyskowej i statystykami i-węzłów. Może być używana do wykrywania niskopoziomowego.
 Wartość zwracana: *ciąg JSON*.
 Obsługiwane platformy: Linux, FreeBSD, Solaris, HP-UX, AIX, MacOS X, OpenBSD, NetBSD, Windows.

Parametry:

- **mode** - możliwe wartości:
  - *full* (domyślnie) - właściwości systemu plików wraz z informacjami o przestrzeni dyskowej i statystykami i-węzłów;
  - *short* - tylko właściwości systemu plików, bez informacji o przestrzeni dyskowej i statystyk i-węzłów;
- **mountpoint** - punkt montowania, dla którego mają zostać zwrócone dane (na przykład `/var` lub `C:`). Jeśli nie zostanie określony, zwracane są wszystkie zamontowane systemy plików.

Uwagi:

- W trybie *short* zwracane są tylko nazwa systemu plików, typ i opcje montowania (w systemie Windows: nazwa, typ, etykieta i typ dysku). Systemy plików nie są odpytywane o statystyki; w systemach uniksopodobnych odczytywana jest tylko lista zamontowanych systemów plików (na przykład `/proc/mounts` w systemie Linux), dlatego sprawdzenie nie wywołuje automatycznego montowania i nie jest zależne od nieodpowiadających sieciowych systemów plików. Dzięki temu tryb *short* nadaje się do wykrywania niskopoziomowego.
- Wartość **mountpoint** musi dokładnie odpowiadać właściwości `fsname`. W systemie Windows porównanie nie uwzględnia wielkości liter; należy określić punkt montowania bez końcowego ukośnika odwrotnego (na przykład `C:`).
- Jeśli określono **mountpoint**, dane są zwracane tylko dla tego punktu montowania. Jeśli nie pasuje do niego żaden zamontowany system plików, zwracana jest pusta tablica (`[]`).
- Dane wyjściowe mogą zawierać kilka wpisów z tą samą wartością `fsname`, jeśli w tym samym katalogu zamontowano więcej niż jeden system plików.
- W trybie *full* informacje o dostępnej przestrzeni dyskowej i statystyki i-węzłów są zwracane tylko wtedy, gdy użytkownik Zabbix agent ma uprawnienia do przechodzenia przez każdy katalog w ścieżce punktu montowania; w przeciwnym razie system plików jest pomijany w danych wyjściowych. Nie dotyczy to trybu *short*, który może zatem wyświetlać systemy plików nieobecne w danych wyjściowych trybu *full*.
- Właściwości `fslabel` i `fsdrivetype` są zwracane tylko w systemie Windows; właściwości `inodes` i `options` nie są zwracane w systemie Windows.
- Raportowane są również systemy plików, w których liczba i-węzłów wynosi zero, co może mieć miejsce w przypadku systemów plików z dynamicznymi i-węzłami (np. btrfs).
- Zobacz także: Wykrywanie zamontowanych systemów plików.

Przykłady:

```
vfs.fs.get #wszystkie zamontowane systemy plików ze statystykami; to samo co vfs.fs.get[full]
vfs.fs.get[short] #wszystkie zamontowane systemy plików bez statystyk
vfs.fs.get[full,/var] #statystyki tylko dla punktu montowania /var
vfs.fs.get[short,C:] #tylko właściwości woluminu C: (Windows)
```

vfs.fs.inode[fs,<mode>]

 Liczba lub wartość procentowa inode'ów.
 Zwracana wartość: *Integer* - dla liczby; *Float* - dla wartości procentowej.
 Zobacz obsługiwane platformy.

Parametry:

- **fs** - system plików;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *used*, *pfree* (wolne, procentowo) lub *pused* (użyte, procentowo).

Jeśli liczba inode'ów jest równa zero, co może mieć miejsce w przypadku systemów plików z dynamicznymi inode'ami (np. btrfs), wartości pfree/pused będą raportowane odpowiednio jako „100" i „0".

Przykład:

```
vfs.fs.inode[/,pfree]
```

vfs.fs.size[fs,<mode>]

 Miejsce na dysku w bajtach lub jako wartość procentowa całkowitej pojemności.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.
 Zobacz obsługiwane platformy.

Parametry:

- **fs** - system plików;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *used*, *pfree* (wolne, procentowo) lub *pused* (użyte, procentowo).

Uwagi:

- Jeśli system plików nie jest zamontowany, zwracany jest rozmiar lokalnego systemu plików, w którym znajduje się punkt montowania.
- Uwzględniana jest zarezerwowana przestrzeń systemu plików i nie jest ona wliczana przy użyciu trybu *free*.

Przykład:

```
vfs.fs.size[/tmp,free]
```

vm.memory.size[<mode>]

 Rozmiar pamięci w bajtach lub jako wartość procentowa całkowitej pamięci.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.
 Zobacz obsługiwane platformy.

Parametr:

- **mode** - możliwe wartości: *total* (domyślnie), *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*, *used*, *pused* (używana, wartość procentowa), *available* lub *pavailable* (dostępna, wartość procentowa).

Komentarze:

- Ta pozycja akceptuje trzy kategorie parametrów:
  - *total* - całkowita ilość pamięci;
  - specyficzne dla platformy typy pamięci: *active*, *anon*, *buffers*, *cached*, *exec*, *file*, *free*, *inactive*, *pinned*, *shared*, *slab*, *wired*;
  - oszacowania na poziomie użytkownika dotyczące ilości używanej i dostępnej pamięci: *used*, *pused*, *available*, *pavailable*.
- Parametr trybu *active* jest obsługiwany tylko w systemach FreeBSD, HP-UX, MacOS X, OpenBSD, NetBSD.
- Parametry trybu *anon*, *exec*, *file* są obsługiwane tylko w systemie NetBSD.
- Parametr trybu *buffers* jest obsługiwany tylko w systemach Linux, FreeBSD, OpenBSD, NetBSD.
- Parametr trybu *cached* jest obsługiwany tylko w systemach Linux, FreeBSD, AIX, OpenBSD, NetBSD.
- Parametry trybu *inactive*, *wired* są obsługiwane tylko w systemach FreeBSD, MacOS X, OpenBSD, NetBSD.
- Parametr trybu *pinned* jest obsługiwany tylko w systemie AIX.
- Parametr trybu *shared* jest obsługiwany tylko w systemach Linux 2.4, FreeBSD, OpenBSD, NetBSD.
- Zobacz także dodatkowe szczegóły dotyczące tej pozycji.

Przykład:

```
vm.memory.size[pavailable]
```

web.page.get[host,<path>,<port>,<redirect_limit>]

 Pobiera zawartość strony internetowej.
 Zwracana wartość: źródło strony internetowej w postaci tekstu (w tym nagłówki).
 Zobacz obsługiwane platformy.

Parametry:

- **host** - nazwa hosta lub URL (w postaci `scheme://host:port/path`, gdzie tylko *host* jest wymagany). Dozwolone schematy URL: *http*, *https*[4]. Brak schematu zostanie potraktowany jako *http*. Jeśli określono URL, wartości `path` i `port` muszą być puste. Podanie nazwy użytkownika/hasła podczas łączenia z serwerami wymagającymi uwierzytelniania, na przykład: `http://user:password@www.example.com`, jest możliwe tylko przy obsłudze cURL [4]. Nazwy hostów obsługują Punycode.
- **path** - ścieżka do dokumentu HTML (domyślnie /);
- **port** - numer portu (domyślnie 80 dla HTTP)

- **redirect_limit** - maksymalna liczba przekierowań, za którymi należy podążać [4], od 0 do 1000 (domyślnie 0 - przekierowania nie są śledzone).

Uwagi:

- Ta pozycja staje się nieobsługiwana, jeśli zasób określony w `host` nie istnieje lub jest niedostępny albo jeśli `redirect_limit` jest większy niż 0 i liczba przekierowań przekracza tę wartość.
- `host` może być nazwą hosta, nazwą domeny, adresem IPv4 lub IPv6. Jednak w przypadku adresu IPv6 agent Zabbix musi zostać skompilowany z włączoną obsługą IPv6.
- Jeśli przekierowania są śledzone, zwracana wartość zawiera nagłówki każdej odpowiedzi przekierowania, a następnie nagłówki i treść odpowiedzi końcowej.

Przykłady:

```
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>]

 Czas ładowania całej strony internetowej (w sekundach).
 Zwracana wartość: *Float*.
 Zobacz obsługiwane platformy.

Parametry:

- **host** - nazwa hosta lub adres URL (w postaci `scheme://host:port/path`, gdzie tylko *host* jest wymagany). Dozwolone schematy URL: *http*, *https*[4]. Brak schematu zostanie potraktowany jako *http*. Jeśli określono adres URL, wartości `path` i `port` muszą być puste. Podanie nazwy użytkownika i hasła podczas łączenia się z serwerami wymagającymi uwierzytelniania, na przykład: `http://user:password@www.example.com`, jest możliwe tylko przy obsłudze cURL [4]. Nazwy hostów zapisane w formacie Punycode są obsługiwane.
- **path** - ścieżka do dokumentu HTML (domyślnie /);
- **port** - numer portu (domyślnie 80 dla HTTP)
- **redirect_limit** - maksymalna liczba przekierowań, za którymi należy podążać [4], od 0 do 1000 (domyślnie 0 - przekierowania nie są śledzone).

Uwagi:

- Ta pozycja przechodzi w stan nieobsługiwanej, jeśli zasób określony w `host` nie istnieje lub jest niedostępny, albo jeśli `redirect_limit` jest większy niż 0 i liczba przekierowań przekracza tę wartość.
- `host` może być nazwą hosta, nazwą domeny, adresem IPv4 lub IPv6. Jednak w przypadku adresu IPv6 agent Zabbix musi zostać skompilowany z włączoną obsługą IPv6.
- Jeśli przekierowania są śledzone, czas ładowania obejmuje czas poświęcony na wszystkie przekierowania.

Przykłady:

```
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>]

 Znajdź ciąg znaków na stronie internetowej.
 Zwracana wartość: dopasowany ciąg znaków lub wartość określona opcjonalnym parametrem `output`.
 Zobacz obsługiwane platformy.

Parametry:

- **host** - nazwa hosta lub URL (w postaci `scheme://host:port/path`, gdzie tylko *host* jest wymagany). Dozwolone schematy URL: *http*, *https*[4]. Brak schematu zostanie potraktowany jako *http*. Jeśli określono URL, parametry `path` i `port` muszą być puste. Podanie nazwy użytkownika i hasła podczas łączenia z serwerami wymagającymi uwierzytelniania, na przykład: `http://user:password@www.example.com`, jest możliwe tylko przy obsłudze cURL [4]. Nazwy hostów zapisane w formacie Punycode są obsługiwane.
- **path** - ścieżka do dokumentu HTML (domyślnie /).
- **port** - numer portu (domyślnie 80 dla HTTP).
- **regexp** - wyrażenie regularne opisujące wymagany wzorzec.
- **length** - maksymalna liczba znaków do zwrócenia;
- **output** - opcjonalny szablon formatowania danych wyjściowych. Sekwencja ucieczki **\0** jest zastępowana dopasowanym fragmentem tekstu (od pierwszego znaku, w którym rozpoczyna się dopasowanie, do znaku, w którym się kończy), natomiast

sekwencja ucieczki **\N** (gdzie N=1...9) jest zastępowana N-tą dopasowaną grupą (lub pustym ciągiem, jeśli N przekracza liczbę przechwyconych grup).
- **redirect_limit** - maksymalna liczba przekierowań, za którymi należy podążać [4], od 0 do 1000 (domyślnie 0 - przekierowania nie są śledzone).

Uwagi:

- Ta pozycja przechodzi w stan nieobsługiwanej, jeśli zasób określony w `host` nie istnieje lub jest niedostępny albo jeśli `redirect_limit` jest większy od 0 i liczba przekierowań przekracza tę wartość.
- `host` może być nazwą hosta, nazwą domeny, adresem IPv4 lub IPv6. Jednak w przypadku adresu IPv6 agent Zabbix musi zostać skompilowany z włączoną obsługą IPv6.
- Wyodrębnianie treści za pomocą parametru `output` odbywa się na agencie.
- Jeśli przekierowania są śledzone, nagłówki każdej odpowiedzi przekierowania również są przeszukiwane.

Przykłady:

```
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

 Metadane hosta agenta.
 Wartość zwracana: *String*.
 Zobacz obsługiwane platformy.

Zwraca wartość parametrów `HostMetadata` lub `HostMetadataItem`, albo pusty ciąg znaków, jeśli nie zdefiniowano żadnego z nich.

agent.hostname

 Nazwa hosta agent.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Zwraca:

- Jako check pasywny - nazwę pierwszego hosta wymienionego w parametrze Hostname pliku konfiguracyjnego agent;
- Jako check aktywny - nazwę bieżącego hosta.

agent.ping

 Sprawdzenie dostępności agenta.
 Zwracana wartość: Nic - niedostępny; 1 - dostępny.
 Zobacz obsługiwane platformy.

Użyj funkcji wyzwalacza `nodata()`, aby sprawdzić niedostępność hosta.

agent.variant

 Wariant agenta Zabbix (Zabbix agent lub Zabbix agent 2).
 Zwracana wartość: 1 - Zabbix agent; 2 - Zabbix agent 2.
 Zobacz obsługiwane platformy.

agent.version

 Wersja agenta Zabbix.
 Zwracana wartość: *String*.
 Zobacz obsługiwane platformy.

Przykład zwracanej wartości:

```
6.0.3
```

zabbix.stats[<ip>,<port>]

 Zwraca zestaw wewnętrznych metryk serwera Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z lokalnej instancji.
 Wartość zwracana: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** - lista adresów IP/DNS/masek sieciowych serwerów/proxy, które mają być zdalnie odpytywane (domyślnie `127.0.0.1`);
- **port** - port serwera/proxy, który ma być zdalnie odpytywany (domyślnie 10051)

Uwagi:

- Ten element zwraca wybrany zestaw wewnętrznych metryk. Szczegóły: zobacz Zdalne monitorowanie statystyk Zabbix.
- Należy pamiętać, że żądanie statystyk będzie akceptowane tylko z adresów wymienionych w parametrze `StatsAllowedIP` serwer/proxy na docelowej instancji.

zabbix.stats[<ip>,<port>,queue,<from>,<to>]

 Zwraca liczbę monitorowanych pozycji w kolejce, które są opóźnione na serwerze Zabbix lub proxy. Jeśli określono `ip` i `port`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z lokalnej instancji.
 Wartość zwracana: *ciąg JSON*.
 Zobacz obsługiwane platformy.

Parametry:

- **ip** - lista adresów IP/DNS/masek sieciowych serwerów/proxy, do których ma zostać wykonane zdalne zapytanie (domyślnie `127.0.0.1`);
- **port** - port serwera/proxy, do którego ma zostać wykonane zdalne zapytanie (domyślnie 10051)
- **queue** - stała (należy użyć bez zmian)
- **from** - opóźnione co najmniej o (domyślnie 6 sekund)
- **to** - opóźnione co najwyżej o (domyślnie nieskończoność)

Należy pamiętać, że żądanie statystyk zostanie zaakceptowane tylko z adresów wymienionych w parametrze 'StatsAllowedIP' serwer/proxy na docelowej instancji.

Przypisy

[1] Uwaga dotycząca systemu Linux.
agent Zabbix musi mieć dostęp tylko do odczytu do systemu plików `/proc`.
Poprawki jądra z witryny www.grsecurity.org ograniczają prawa dostępu użytkowników bez uprawnień uprzywilejowanych.

[2] `vfs.dev.read[]`, `vfs.dev.write[]`: agent Zabbix zakończy „nieaktualne" połączenia z urządzeniami, jeśli wartości pozycji nie będą odczytywane przez ponad 3 godziny.
Może się tak zdarzyć, jeśli system zawiera urządzenia o dynamicznie zmieniających się ścieżkach lub jeśli urządzenie zostanie ręcznie usunięte.
Należy również pamiętać, że te pozycje, jeśli używają interwału aktualizacji wynoszącego 3 godziny lub więcej, zawsze zwrócą wartość „0".

[3] `vfs.dev.read[]`, `vfs.dev.write[]`: Jeśli dla pierwszego parametru użyto domyślnej wartości `all`, klucz zwróci statystyki podsumowujące, obejmujące wszystkie urządzenia blokowe, takie jak sda i sdb, oraz ich partycje (sda1, sda2, sdb3...), a także wiele urządzeń (macierz MD RAID) opartych na tych urządzeniach blokowych/partycjach i woluminy logiczne (LVM) oparte na tych urządzeniach blokowych/partycjach.
W takich przypadkach zwracane wartości należy traktować wyłącznie jako wartości względne (dynamiczne w czasie), a nie jako wartości bezwzględne.

[4] Protokół HTTPS i obsługa przekierowań (wartość `redirect_limit` większa niż 0) są obsługiwane tylko wtedy, gdy agent Zabbix został skompilowany z obsługą cURL; w przeciwnym razie pozycja stanie się nieobsługiwana. Zabbix agent 2 nie wymaga cURL w żadnym z tych przypadków.

[5] Wartości `bytes` i `errors` nie są obsługiwane dla interfejsów loopback w systemach Solaris do wersji Solaris 10 6/06 włącznie, ponieważ statystyki bajtów, błędów i wykorzystania nie są przechowywane i/lub raportowane przez jądro.
Jeśli jednak monitorujesz system Solaris za pośrednictwem net-snmp, wartości mogą być zwracane, ponieważ net-snmp zawiera starszy kod pochodzący z cmu-snmp, datowany nawet na 1997 rok, który w przypadku niepowodzenia odczytu wartości bajtów ze statystyk interfejsu zwraca licznik pakietów (który istnieje dla interfejsów loopback) pomnożony przez arbitralną wartość 308.
Zakłada to, że średnia długość pakietu wynosi 308 oktetów, co jest bardzo przybliżonym oszacowaniem, ponieważ limit MTU w systemach Solaris dla interfejsów loopback wynosi 8892 bajty.
Nie należy zakładać, że te wartości są poprawne ani nawet w przybliżeniu dokładne.
Są to wartości szacunkowe.
agent Zabbix nie wykonuje takich przybliżonych obliczeń, ale net-snmp zwróci wartość dla tych pól.

[6] Wiersz poleceń w systemie Solaris, uzyskiwany z `/proc/pid/psinfo`, jest ograniczony do 80 bajtów i zawiera wiersz poleceń w postaci z chwili uruchomienia procesu.

[7] Pozycje `vfs.file.contents[]`, `vfs.file.regexp[]`, `vfs.file.regmatch[]` mogą służyć do pobierania zawartości plików. Jeśli chcesz ograniczyć dostęp do określonych plików zawierających poufne informacje, uruchom agenta Zabbix jako użytkownik, który nie ma uprawnień do wyświetlania tych plików.

Użycie z narzędziami wiersza poleceń

Należy pamiętać, że podczas testowania lub używania kluczy pozycji za pomocą `zabbix_agentd` lub `zabbix_get` z wiersza poleceń należy również uwzględnić składnię powłoki.

Na przykład, jeśli określony parametr klucza musi być ujęty w podwójne cudzysłowy, należy jawnie poprzedzić podwójne cudzysłowy znakiem ucieczki, w przeciwnym razie zostaną one usunięte przez powłokę jako znaki specjalne i nie zostaną przekazane do narzędzia Zabbix.

Przykłady:

```
zabbix_agentd -t 'vfs.dir.count[/var/log,,,"file,dir",,0]'

zabbix_agentd -t vfs.dir.count[/var/log,,,\"file,dir\",,0]
```

Ustawienia kodowania

Aby upewnić się, że pozyskane dane nie zostaną uszkodzone, można określić prawidłowe kodowanie używane podczas przetwarzania sprawdzenia (np. `vfs.file.contents`) w parametrze `encoding`. Lista obsługiwanych kodowań (identyfikatorów stron kodowych) znajduje się w dokumentacji libiconv (GNU Project) lub w dokumentacji Microsoft Windows SDK dotyczącej "Code Page Identifiers". Należy pamiętać, że firma Microsoft czasami oznacza niektóre strony kodowe jako „dostępne tylko dla aplikacji zarządzanych", co oznacza, że są one udostępniane wyłącznie za pośrednictwem środowiska uruchomieniowego .NET i nie są dostępne za pośrednictwem natywnych interfejsów API Win32. agent Zabbix implementuje własną logikę konwersji kodowania, dlatego te strony kodowe są obsługiwane przez agent Zabbix nawet wtedy, gdy natywne funkcje systemu Windows ich nie udostępniają.

Jeśli w parametrze `encoding` nie określono kodowania, stosowane są następujące strategie ustalania kodowania:

- Jeśli kodowanie nie zostało określone (lub jest pustym ciągiem), przyjmuje się UTF-8, a dane są przetwarzane „bez zmian";
- Analiza BOM - dotyczy pozycji `vfs.file.contents`, `vfs.file.regexp`, `vfs.file.regmatch`. Podejmowana jest próba określenia prawidłowego kodowania na podstawie znacznika kolejności bajtów (BOM) znajdującego się na początku pliku. Jeśli BOM nie jest obecny, stosowana jest standardowa strategia ustalania kodowania (zobacz powyżej).

Rozwiązywanie problemów z pozycjami agenta

W przypadku kontroli pasywnych, aby zapobiec sytuacji, w której pozycja nie otrzyma żadnej wartości, ponieważ żądanie serwera do agenta zakończy się przekroczeniem limitu czasu jako pierwsze, należy uwzględnić następujące kwestie:

- Jeśli wersja agenta jest starsza niż wersja serwera, wartość `Timeout` w konfiguracji pozycji (lub globalny limit czasu) może wymagać ustawienia na wyższą niż wartość `Timeout` w pliku konfiguracyjnym agenta.
- Jeśli wersja agenta jest nowsza niż wersja serwera, wartość `Timeout` w pliku konfiguracyjnym serwera może wymagać ustawienia na wyższą niż wartość `Timeout` w pliku konfiguracyjnym agenta.

Zabbix agent 2

Zabbix agent 2 obsługuje wszystkie klucze pozycji obsługiwane przez Zabbix agent w Unix i Windows. Ta strona zawiera szczegóły dodatkowych kluczy pozycji, których można używać wyłącznie z Zabbix agent 2, pogrupowanych według wtyczki, do której należą.

Klucze pozycji są wymienione bez parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Klucz pozycji	Opis	Wtyczka
ceph.df.details	Użycie danych klastra i ich rozmieszczenie między pulami.	Ceph (ładowalna)
ceph.osd.stats	Zagregowane statystyki oraz statystyki dla poszczególnych OSD.	
ceph.osd.discovery	Lista wykrytych OSD.	
ceph.osd.dump	Progi użycia i stany OSD.	
ceph.ping	Testuje, czy można nawiązać połączenie z Ceph.	
ceph.pool.discovery	Lista wykrytych pul.	
ceph.status	Ogólny stan klastra.	
docker.container_info	Informacje niskiego poziomu o kontenerze.	Docker
docker.container_stats	Statystyki użycia zasobów kontenera.	
docker.containers	Zwraca listę kontenerów.	
docker.containers.discovery	Zwraca listę kontenerów. Używane do niskopoziomowego wykrywania.	
docker.data.usage	Informacje o bieżącym użyciu danych.	
docker.images	Zwraca listę obrazów.	
docker.images.discovery	Zwraca listę obrazów. Używane do niskopoziomowego wykrywania.	
docker.info	Informacje o systemie.	
docker.ping	Testuje, czy demon Docker działa.	
ember.get	Zwraca wynik wymaganego urządzenia.	Ember+ (ładowalna)
memcached.ping	Testuje, czy połączenie jest aktywne.	Memcached
memcached.stats	Pobiera wynik polecenia STATS.	
mongodb.cfg.discovery	Zwraca listę wykrytych serwerów konfiguracji.	MongoDB (ładowalna)
mongodb.collection.stats	Zwraca różne statystyki magazynu dla danej kolekcji.	
mongodb.collections.discovery	Zwraca listę wykrytych kolekcji.	
mongodb.collections.usage	Zwraca statystyki użycia dla kolekcji.	
mongodb.connpool.stats	Zwraca informacje dotyczące otwartych połączeń wychodzących z bieżącej instancji bazy danych do innych członków klastra shardowanego lub zestawu replik.	
mongodb.db.stats	Zwraca statystyki odzwierciedlające stan danej bazy danych.	

Klucz pozycji	Opis	Wtyczka
mongodb.db.discovery	Zwraca listę wykrytych baz danych.	
mongodb.jumbo_chunks.count	Zwraca liczbę jumbo chunks.	
mongodb.oplog.stats	Zwraca stan zestawu replik na podstawie danych odczytanych z oplog.	
mongodb.ping	Testuje, czy połączenie jest aktywne.	
mongodb.rs.config	Zwraca bieżącą konfigurację zestawu replik.	
mongodb.rs.status	Zwraca stan zestawu replik z punktu widzenia członka, na którym uruchomiono metodę.	
mongodb.server.status	Zwraca stan bazy danych.	
mongodb.sh.discovery	Zwraca listę wykrytych shardów obecnych w klastrze.	
mongodb.version	Zwraca wersję serwera bazy danych.	
mqtt.get	Subskrybuje określony temat lub tematy (z symbolami wieloznacznymi) podanego brokera i oczekuje na publikacje.	MQTT
mssql.availability.group.get	Zwraca grupy dostępności.	MSSQL (ładowalna)
mssql.custom.query	Zwraca wynik zapytania niestandardowego.	
mssql.db.get	Zwraca wszystkie dostępne bazy danych MSSQL.	
mssql.job.status.get	Zwraca stan zadań.	
mssql.last.backup.get	Zwraca czas ostatniej kopii zapasowej dla wszystkich baz danych.	
mssql.local.db.get	Zwraca bazy danych uczestniczące w grupie dostępności Always On i replikacji (primary lub secondary), które znajdują się na serwerze, z którym nawiązano połączenie.	
mssql.mirroring.get	Zwraca informacje o mirroringu.	
mssql.nonlocal.db.get	Zwraca bazy danych uczestniczące w grupie dostępności Always On i replikacji (primary lub secondary), znajdujące się na innych serwerach (baza danych nie jest lokalna względem instancji SQL Server, z którą nawiązano połączenie).	
mssql.perfcounter.get	Zwraca liczniki wydajności.	
mssql.ping	Testuje, czy połączenie jest aktywne.	
mssql.quorum.get	Zwraca informacje o quorum.	
mssql.quorum.member.get	Zwraca członków quorum.	
mssql.replica.get	Zwraca repliki.	
mssql.version	Zwraca wersję MSSQL.	
mysql.custom.query	Zwraca wynik zapytania niestandardowego.	MySQL
mysql.db.discovery	Zwraca listę baz danych MySQL.	
mysql.db.size	Rozmiar bazy danych w bajtach.	
mysql.get_status_variables	Wartości globalnych zmiennych stanu.	
mysql.ping	Testuje, czy połączenie jest aktywne.	
mysql.replication.discovery	Zwraca listę replikacji MySQL.	
mysql.replication.get_slave_status	Stan replikacji.	
mysql.version	Wersja MySQL.	
net.dns.get	Wykonuje zapytanie DNS i zwraca szczegółowe informacje o rekordzie DNS.	Network
nvml.device.count	Liczba urządzeń GPU.	NVIDIA GPU (ładowalna)
nvml.device.decoder.utilization	Wykorzystanie dekodera urządzenia GPU.	
nvml.device.ecc.mode	Tryb ECC urządzenia GPU.	
nvml.device.encoder.stats.get	Statystyki enkodera urządzenia GPU.	
nvml.device.encoder.utilization	Wykorzystanie enkodera urządzenia GPU.	
nvml.device.energy.consumption	Zużycie energii przez urządzenie GPU.	
nvml.device.errors.memory	Informacje o błędach pamięci ECC urządzenia GPU.	
nvml.device.errors.register	Informacje o błędach rejestru ECC urządzenia GPU.	
nvml.device.fan.speed.avg	Średnia prędkość wentylatora urządzenia GPU.	
nvml.device.get	Wykryte GPU z identyfikatorem UUID i nazwą.	
nvml.device.graphics.frequency	Częstotliwość zegara grafiki urządzenia GPU.	
nvml.device.memory.bar1.get	Informacje o pamięci BAR1 urządzenia GPU.	
nvml.device.memory.fb.get	Informacje o pamięci framebuffer urządzenia GPU.	
nvml.device.memory.frequency	Częstotliwość zegara pamięci urządzenia GPU.	
nvml.device.pci.utilization	Wykorzystanie PCI urządzenia GPU.	
nvml.device.performance.state	Stan wydajności urządzenia GPU.	
nvml.device.power.limit	Limit mocy urządzenia GPU.	
nvml.device.power.usage	Zużycie mocy urządzenia GPU.	
nvml.device.serial	Numer seryjny urządzenia GPU.	

Klucz pozycji	Opis	Wtyczka
nvml.device.sm.frequency	Częstotliwość zegara multiprocesora strumieniowego urządzenia GPU.	
nvml.device.temperature	Temperatura urządzenia GPU.	
nvml.device.utilization	Statystyki wykorzystania urządzenia GPU.	
nvml.device.video.frequency	Częstotliwość zegara wideo urządzenia GPU.	
nvml.system.driver.version	Wersja sterownika NVIDIA.	
nvml.version	Wersja biblioteki NVML.	
oracle.diskgroups.stats	Zwraca statystyki grup dysków Automatic Storage Management (ASM).	Oracle
oracle.diskgroups.discovery	Zwraca listę grup dysków ASM.	
oracle.archive.info	Statystyki dzienników archiwalnych.	
oracle.cdb.info	Informacje o Container Databases (CDB).	
oracle.custom.query	Wynik zapytania niestandardowego.	
oracle.datafiles.stats	Zwraca statystyki plików danych.	
oracle.db.discovery	Zwraca listę baz danych.	
oracle.fra.stats	Zwraca statystyki Fast Recovery Area (FRA).	
oracle.instance.info	Statystyki instancji.	
oracle.pdb.info	Informacje o Pluggable Databases (PDB).	
oracle.pdb.discovery	Zwraca listę PDB.	
oracle.pga.stats	Zwraca statystyki Program Global Area (PGA).	
oracle.ping	Testuje, czy można nawiązać połączenie z Oracle.	
oracle.proc.stats	Zwraca statystyki procesów.	
oracle.redolog.info	Informacje o plikach dziennika z pliku kontrolnego.	
oracle.sga.stats	Zwraca statystyki System Global Area (SGA).	
oracle.sessions.stats	Zwraca statystyki sesji.	
oracle.sys.metrics	Zwraca zestaw wartości metryk systemowych.	
oracle.sys.params	Zwraca zestaw wartości parametrów systemowych.	
oracle.ts.stats	Zwraca statystyki tablespace’ów.	
oracle.ts.discovery	Zwraca listę tablespace’ów.	
oracle.user.info	Zwraca informacje o użytkowniku Oracle.	
oracle.version	Zwraca wersję serwera bazy danych.	
pgsql.autovacuum.count	Liczba procesów roboczych autovacuum.	PostgreSQL (ładowalna)
pgsql.archive	Informacje o zarchiwizowanych plikach.	
pgsql.bgwriter	Łączna liczba checkpointów dla klastra baz danych, z podziałem według typu checkpointu.	
pgsql.cache.hit	Współczynnik trafień w bufor cache PostgreSQL.	
pgsql.connections	Zwraca połączenia według typu.	
pgsql.custom.query	Zwraca wynik zapytania niestandardowego.	
pgsql.db.age	Wiek najstarszego FrozenXID bazy danych.	
pgsql.db.bloating_tables	Liczba tabel z nadmiernym przyrostem rozmiaru w każdej bazie danych.	
pgsql.db.discovery	Lista baz danych PostgreSQL.	
pgsql.db.size	Rozmiar bazy danych w bajtach.	
pgsql.dbstat	Zbiera statystyki dla każdej bazy danych.	
pgsql.dbstat.sum	Zestawione dane dla wszystkich baz danych w klastrze.	
pgsql.locks	Informacje o przyznanych blokadach dla każdej bazy danych.	
pgsql.oldest.xid	Wiek najstarszego XID.	
pgsql.ping	Testuje, czy połączenie jest aktywne.	
pgsql.queries	Metryki zapytań według czasu wykonania.	
pgsql.replication.count	Liczba serwerów standby.	
pgsql.replication.process	Opóźnienie flush, opóźnienie write i opóźnienie replay dla każdego procesu wysyłającego.	
pgsql.replication.process.discovery	Wykrywanie nazw procesów replikacji.	
pgsql.replication.recovery_role	Stan odzyskiwania.	
pgsql.replication.status	Stan replikacji.	
pgsql.replication_lag.b	Opóźnienie replikacji w bajtach.	
pgsql.replication_lag.sec	Opóźnienie replikacji w sekundach.	
pgsql.uptime	Czas działania PostgreSQL w milisekundach.	
pgsql.version	Zwraca wersję PostgreSQL.	
pgsql.wal.stat	Statystyki WAL.	
redis.config	Pobiera parametry konfiguracji instancji Redis pasujące do wzorca.	Redis
redis.info	Pobiera wynik polecenia INFO.	
redis.ping	Testuje, czy połączenie jest aktywne.	

Klucz pozycji	Opis	Wtyczka
redis.slowlog.count	Liczba wpisów w dzienniku wolnych operacji od uruchomienia Redis.	
smart.attribute.discovery	Zwraca listę atrybutów urządzeń S.M.A.R.T.	S.M.A.R.T.
smart.disk.discovery	Zwraca listę urządzeń S.M.A.R.T.	
smart.disk.get	Zwraca wszystkie dostępne właściwości urządzeń S.M.A.R.T.	
systemd.unit.get	Zwraca wszystkie właściwości jednostki systemd.	Systemd
systemd.unit.info	Informacje o jednostce systemd.	
systemd.unit.discovery	Lista jednostek systemd i ich szczegóły.	
web.certificate.get	Weryfikuje certyfikaty i zwraca szczegóły certyfikatu.	Web certificates

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

ceph.df.details[connString,<user>,<apikey>]

 Wykorzystanie danych klastra i ich dystrybucja pomiędzy pulami.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.osd.stats[connString,<user>,<apikey>]

 Statystyki zagregowane i dla poszczególnych OSD.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, apikey** - nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.osd.discovery[connString,<user>,<apikey>]

 Lista wykrytych OSD. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.osd.dump[connString,<user>,<apikey>]

 Progi wykorzystania i statusy OSD.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, apikey** - nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.ping[connString,<user>,<apikey>]

 Sprawdza, czy można nawiązać połączenie z Ceph.
 Wartość zwracana: *0* - połączenie jest przerwane (jeśli występuje jakikolwiek błąd, w tym problemy z AUTH i konfiguracją); *1* - połączenie zostało nawiązane pomyślnie.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, apikey** - nazwa użytkownika i klucz API do łączenia z Ceph.

ceph.pool.discovery[connString,<user>,<apikey>]

 Lista wykrytych puli. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

ceph.status[connString,<user>,<apikey>]

 Ogólny status klastra.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, apikey** — nazwa użytkownika i klucz API do połączenia z Ceph.

docker.container_info[<ID>,<info>]

 Informacje niskiego poziomu o kontenerze.
 Zwracana wartość: Dane wyjściowe wywołania API ContainerInspect zserializowane jako JSON.

Parametry:

- **ID** - identyfikator lub nazwa kontenera;
- **info** - ilość zwracanych informacji. Obsługiwane wartości: *short* (domyślnie) lub *full*.

Użytkownik Agent 2 (’zabbix’) musi zostać dodany do grupy ’docker’, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.container_stats[<ID>]

 Statystyki użycia zasobów kontenera.
 Wartość zwracana: dane wyjściowe wywołania API ContainerStats oraz procent użycia CPU zserializowane jako JSON.

Parametr:

- **ID** — identyfikator lub nazwa kontenera.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby mieć wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.containers

 Lista kontenerów.
 Zwracana wartość: Dane wyjściowe wywołania API ContainerList zserializowane jako JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do group „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.containers.discovery[<options>]

 Zwraca listę kontenerów. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **options** — określa, czy mają zostać wykryte wszystkie kontenery, czy tylko uruchomione. Obsługiwane wartości: *true* — zwraca wszystkie kontenery; *false* — zwraca tylko uruchomione kontenery (domyślnie).

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.data.usage

 Informacje o bieżącym użyciu danych.
 Zwracana wartość: Dane wyjściowe wywołania API SystemDataUsage zserializowane jako JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do group „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.images

 Zwraca listę obrazów.
 Wartość zwracana: dane wyjściowe wywołania API ImageList zserializowane do formatu JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.images.discovery

 Zwraca listę obrazów. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.info

 Informacje o systemie.
 Wartość zwracana: Dane wyjściowe wywołania API SystemInfo zserializowane jako JSON.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby miał wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

docker.ping

 Sprawdza, czy demon Docker działa, czy nie.
 Zwracana wartość: *1* - połączenie działa; *0* - połączenie jest zerwane.

Użytkownik Agent 2 („zabbix”) musi zostać dodany do grupy „docker”, aby mieć wystarczające uprawnienia. W przeciwnym razie sprawdzenie zakończy się niepowodzeniem.

ember.get[<uri>,<path>]

 Zwraca wynik wymaganego urządzenia.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI urządzenia Ember+. Domyślnie: 127.0.0.1:9998;
- **path** - ścieżka OID do urządzenia. Domyślnie pusta, zwraca dane kolekcji głównej.

memcached.ping[connString,<user>,<password>]

 Sprawdza, czy połączenie jest aktywne.
 Zwracana wartość: *1* — połączenie jest aktywne; *0* — połączenie jest zerwane (jeśli wystąpi dowolny błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do Memcached.

memcached.stats[connString,<user>,<password>,<type>]

 Pobiera dane wyjściowe polecenia STATS.
 Wartość zwracana: *JSON* - dane wyjściowe są serializowane do formatu JSON.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do Memcached;
- **type** - typ statystyk do zwrócenia: *items*, *sizes*, *slabs* lub *settings* (domyślnie pusty, zwraca statystyki ogólne).

mongodb.collection.stats[connString,<user>,<password>,<database>,collection]

 Zwraca różne statystyki pamięci masowej dla danej kolekcji.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB;
- **database** - nazwa bazy danych (domyślnie: admin);
- **collection** - nazwa kolekcji.

mongodb.cfg.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych serwerów konfiguracji.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.collections.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych kolekcji. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.collections.usage[connString,<user>,<password>]

 Zwraca statystyki użycia dla kolekcji.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.connpool.stats[connString,<user>,<password>]

 Zwraca informacje dotyczące otwartych połączeń wychodzących z bieżącej instancji bazy danych do innych członków klastra shardingu lub zestawu replik.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.db.stats[connString,<user>,<password>,<database>]

 Zwraca statystyki odzwierciedlające stan danego systemu bazy danych.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB;
- **database** - nazwa bazy danych (domyślnie: admin).

mongodb.db.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych baz danych. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.jumbo_chunks.count[connString,<user>,<password>]

 Zwraca liczbę jumbo chunks.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.oplog.stats[connString,<user>,<password>]

 Zwraca stan zestawu replik, używając danych pobranych z oplog.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.ping[connString,<user>,<password>]

 Sprawdza, czy połączenie jest aktywne, czy nie.
 Zwracana wartość: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.rs.config[connString,<user>,<password>]

 Zwraca bieżącą konfigurację zestawu replik.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mongodb.rs.status[connString,<user>,<password>]

 Zwraca status zestawu replik z punktu widzenia członka, na którym uruchomiono tę metodę.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.server.status[connString,<user>,<password>]

 Zwraca stan bazy danych.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.sh.discovery[connString,<user>,<password>]

 Zwraca listę wykrytych shardów obecnych w klastrze.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MongoDB.

mongodb.version[connString,<user>,<password>]

 Zwraca wersję serwera bazy danych.
 Wartość zwracana: *String*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MongoDB.

mqtt.get[<broker url>,topic,<user>,<password>]

 Subskrybuje określony temat lub tematy (z użyciem symboli wieloznacznych) w podanym brokerze i oczekuje na publikacje.
 Wartość zwracana: Zależnie od zawartości tematu. Jeśli używane są symbole wieloznaczne, zwraca zawartość tematu jako JSON.

Parametry:

- **broker url** - adres URL brokera MQTT w formacie `protocol://host:port` bez parametrów zapytania (obsługiwane protokoły: `tcp`, `ssl`, `ws`). Jeśli nie zostanie podana żadna wartość, agent użyje `tcp://localhost:1883`. Jeśli protokół lub port zostaną pominięte, zostanie użyty domyślny protokół (`tcp`) lub port (1883);
- **topic** - temat MQTT (wymagany). Obsługiwane są symbole wieloznaczne (+,#);
- **user, password** - dane uwierzytelniające (jeśli są wymagane).

Uwagi:

- pozycja musi być skonfigurowana jako active check (typ pozycji „Zabbix agent (active)”);
- certyfikaty szyfrowania TLS można używać, zapisując je w domyślnej lokalizacji (np. katalog `/etc/ssl/certs/` w Ubuntu). Dla TLS użyj schematu `tls://`.

mssql.availability.group.get[URI,<user>,<password>]

 Zwraca grupy dostępności.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **URI** — URI serwera MSSQL (jedynym obsługiwanym schematem jest `sqlserver://`). Osadzone dane uwierzytelniające zostaną zignorowane. Można określić nazwę instancji jako część URI, np.: `sqlserver://localhost/InstanceName` (bez portu). Jeśli określono port, nazwa instancji jest ignorowana;
- **user, password** — nazwa użytkownika, hasło wysyłane do chronionego serwera MSSQL.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.custom.query[URI,<user>,<password>,queryName,<args...>]

 Zwraca wynik niestandardowego zapytania.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **URI** - URI serwera MSSQL (jedynym obsługiwanym schematem jest `sqlserver://`). Osadzone dane uwierzytelniające zostaną zignorowane. Można określić nazwę instancji jako część URI, np.: `sqlserver://localhost/InstanceName` (bez portu). Jeśli określono port, nazwa instancji jest ignorowana;
- **user, password** - nazwa użytkownika, hasło wysyłane do chronionego serwera MSSQL;
- **queryName** - nazwa niestandardowego zapytania skonfigurowanego w `Plugins.MSSQL.CustomQueriesDir` bez rozszerzenia .sql;
- **args** - jeden lub kilka argumentów oddzielonych przecinkami, przekazywanych do zapytania.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.db.get

 Zwraca wszystkie dostępne bazy danych MSSQL.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.job.status.get

 Zwraca status zadań.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.last.backup.get

 Zwraca czas ostatniej kopii zapasowej dla wszystkich baz danych.
 Zwracana wartość: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.local.db.get

 Zwraca bazy danych, które uczestniczą w grupie dostępności Always On i replice (podstawowej lub pomocniczej) oraz znajdują się na serwerze, z którym ustanowiono połączenie.
 Zwracana wartość: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.mirroring.get

 Zwraca informacje o mirroringu.
 Zwracana wartość: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.nonlocal.db.get

 Zwraca bazy danych uczestniczące w grupie dostępności Always On oraz replice (podstawowej lub pomocniczej) znajdującej się na innych serwerach (baza danych nie jest lokalna dla instancji SQL Server, z którą nawiązano połączenie).
 Zwracana wartość: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.perfcounter.get

 Zwraca liczniki wydajności.
 Zwracana wartość: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.ping

 Sprawdza dostępność bazy danych. Testuje, czy połączenie jest poprawnie skonfigurowane.
 Zwracana wartość: *1* - działa, *0* - nie działa.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.quorum.get

 Zwraca informacje o kworum.
 Zwracana wartość: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.quorum.member.get

 Zwraca członków kworum.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.replica.get

 Zwraca repliki.
 Wartość zwracana: *obiekt JSON*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mssql.version

 Zwraca wersję MSSQL.
 Zwracana wartość: *String*.

Więcej informacji można znaleźć w pliku readme wtyczki MSSQL.

mysql.custom.query[connString,<user>,<password>,queryName,<args...>]

 Zwraca wynik niestandardowego zapytania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania MySQL;
- **queryName** - nazwa niestandardowego zapytania; musi odpowiadać nazwie pliku SQL bez rozszerzenia;
- **args** - jeden lub kilka argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Aby uzyskać więcej informacji, zobacz plik readme wtyczki MySQL, sekcję *Niestandardowe zapytania*.

mysql.db.discovery[connString,<user>,<password>]

 Zwraca listę baz danych MySQL. Używane w odkrywaniu niskiego poziomu.
 Wartość zwracana: wynik zapytania SQL „show databases” w formacie JSON LLD.

Parametry:

- **connString** — URI lub nazwa sesji;

- **user, password** — dane logowania do MySQL.

mysql.db.size[connString,<user>,<password>,<database name>]

 Rozmiar bazy danych w bajtach.
 Zwracana wartość: wynik zapytania SQL „select coalesce(sum(data_length + index_length),0) as size from information_schema.tables where table_schema=?” dla określonej bazy danych, w bajtach.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania MySQL;
- **database name** — nazwa bazy danych.

mysql.get_status_variables[connString,<user>,<password>]

 Wartości globalnych zmiennych statusu.
 Wartość zwracana: Wynik zapytania SQL „show global status” w formacie JSON.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania MySQL.

mysql.ping[connString,<user>,<password>]

 Sprawdza, czy połączenie jest aktywne.
 Zwracana wartość: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MySQL.

mysql.replication.discovery[connString,<user>,<password>]

 Zwraca listę replikacji MySQL. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: wynik zapytania SQL `SHOW SLAVE STATUS` lub `SHOW REPLICA STATUS` w formacie LLD JSON.

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania MySQL.

mysql.replication.get_slave_status[connString,<user>,<password>,<master host>]

 Status replikacji.
 Zwracana wartość: wynik zapytania SQL `SHOW SLAVE STATUS` lub `SHOW REPLICA STATUS` w formacie JSON.

Parametry:

- **connString** — URI lub nazwa sesji;
- **user, password** — dane logowania do MySQL;
- **master host** — nazwa hosta głównego replikacji. Jeśli nie znaleziono żadnego, zwracany jest błąd. Jeśli ten parametr nie jest określony, zwracane są wszystkie hosty.

mysql.version[connString,<user>,<password>]

 Wersja MySQL.
 Zwracana wartość: *String* (z wersją instancji MySQL).

Parametry:

- **connString** - URI lub nazwa sesji;
- **user, password** - dane logowania do MySQL.

net.dns.get[<ip>,name,<type>,<timeout>,<count>,<protocol>,”<flags>”]

Wykonuje zapytanie DNS i zwraca szczegółowe informacje o rekordzie DNS.
 Ta pozycja jest rozszerzoną wersją pozycji `net.dns.record` agenta Zabbix z obsługą większej liczby typów rekordów i konfigurowalnych flag.
 Zwracane wartości: *obiekt JSON*

Parametry:

- **ip** - adres IP serwera DNS (pozostaw puste, aby użyć domyślnego serwera DNS);
- **name** - nazwa DNS do zapytania;
- **type** - typ rekordu, o który ma zostać wykonane zapytanie (domyślnie *SOA*);
- **timeout** - limit czasu żądania w sekundach (domyślnie 1 sekunda);
- **count** - liczba prób dla żądania (domyślnie 2);
- **protocol** - protokół używany do wykonywania zapytań DNS: *udp* (domyślnie) lub *tcp*;

- **flags** - jeden lub więcej argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Uwagi:

- Możliwe wartości dla `type` to: *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*. Zwróć uwagę, że wartości muszą być podane wyłącznie wielkimi literami; małe litery lub mieszana wielkość liter nie są obsługiwane.
- W przypadku odwrotnych wyszukiwań DNS (gdy `type` jest ustawiony na *PTR*), można podać nazwę DNS zarówno w formacie odwróconym, jak i nieodwróconym (patrz przykłady poniżej). Zwróć uwagę, że gdy żądany jest rekord PTR, nazwa DNS jest w rzeczywistości adresem IP.
- Możliwe wartości dla `flags` to:
  - *cdflag* lub *nocdflag* (domyślnie);
  - *rdflag* (domyślnie) lub *nordflag*;
  - *dnssec* lub *nodnssec* (domyślnie);
  - *nsid* lub *nonsid* (domyślnie);
  - *edns0* (domyślnie) lub *noedns0*;
  - *aaflag* lub *noaaflag* (domyślnie);
  - *adflag* lub *noadflag* (domyślnie). `flags` *dnssec* i *nsid* nie mogą być używane razem z *noedns0*, ponieważ oba wymagają *edns0*. Zwróć uwagę, że wartości muszą być podane wyłącznie małymi literami; wielkie litery lub mieszana wielkość liter nie są obsługiwane.
- Zinternacjonalizowane nazwy domen nie są obsługiwane, użyj zamiast nich nazw zakodowanych w IDNA.
- Wynik jest obiektem zawierającym informacje o rekordzie DNS na podstawie podanych parametrów (zobacz więcej szczegółów).

Przykłady:

```
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

 Liczba urządzeń GPU.
 Zwracana wartość: *Integer*.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.decoder.utilization[<deviceUUID>]

 Wykorzystanie dekodera urządzenia GPU w procentach.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Aby uzyskać więcej informacji, zobacz plik readme wtyczki NVIDIA GPU.

nvml.device.ecc.mode[<deviceUUID>]

 Informacje o trybie ECC urządzenia GPU (bieżący, oczekujący).
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.encoder.stats.get[<deviceUUID>]

 Statystyki enkodera urządzenia GPU.
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.encoder.utilization[<deviceUUID>]

 Wykorzystanie enkodera urządzenia GPU w procentach.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.energy.consumption[<deviceUUID>]

 Całkowite zużycie energii urządzenia GPU w milidżulach (mJ) od ostatniego przeładowania sterownika.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.errors.memory[<deviceUUID>]

 Informacje o błędach pamięci ECC urządzenia GPU (skorygowane, nieskorygowane).
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.errors.register[<deviceUUID>]

 Informacje o błędach rejestru ECC urządzenia GPU (skorygowane, nieskorygowane).
 Wartość zwracana: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Aby uzyskać więcej informacji, zobacz plik readme wtyczki NVIDIA GPU.

nvml.device.fan.speed.avg[<deviceUUID>]

 Średnia prędkość wentylatora urządzenia GPU jako wartość procentowa maksymalnej prędkości.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.get

 Wykryte procesory GPU z UUID i nazwą.
 Zwracana wartość: *JSON*.

Aby uzyskać więcej informacji, zobacz plik readme wtyczki NVIDIA GPU.

nvml.device.graphics.frequency[<deviceUUID>]

 Częstotliwość zegara grafiki urządzenia GPU w MHz.
 Wartość zwracana: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.memory.fb.get[<deviceUUID>]

 Statystyki pamięci framebuffer urządzenia GPU (całkowita, zarezerwowana, wolna, używana).
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Aby uzyskać więcej informacji, zobacz plik readme wtyczki NVIDIA GPU.

nvml.device.memory.bar1.get[<deviceUUID>]

 Statystyki pamięci BAR1 urządzenia GPU (całkowita, wolna, używana).
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.memory.frequency[<deviceUUID>]

 Częstotliwość taktowania pamięci urządzenia GPU w MHz.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.pci.utilization[<deviceUUID>]

 Wykorzystanie PCI urządzenia GPU (przepustowość nadawania/odbioru w KBps).
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Aby uzyskać więcej informacji, zobacz plik readme wtyczki NVIDIA GPU.

nvml.device.performance.state[<deviceUUID>]

 Stan wydajności urządzenia GPU.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.power.limit[<deviceUUID>]

 Limit mocy urządzenia GPU w miliwatach.
 Zwracana wartość: *Liczba całkowita*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.power.usage[<deviceUUID>]

 Bieżące zużycie energii urządzenia GPU w miliwatach.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.serial[<deviceUUID>]

 Numer seryjny urządzenia GPU.
 Zwracana wartość: *String*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.sm.frequency[<deviceUUID>]

 Częstotliwość zegara multiprocesora strumieniowego urządzenia GPU w MHz.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.temperature[<deviceUUID>]

 Temperatura urządzenia GPU w stopniach Celsjusza.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.utilization[<deviceUUID>]

 Statystyki wykorzystania urządzenia GPU (wykorzystanie GPU/pamięci jako wartość procentowa).
 Zwracana wartość: *JSON*.

Parametr:

- **deviceUUID** — UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.device.video.frequency[<deviceUUID>]

 Częstotliwość zegara wideo urządzenia GPU w MHz.
 Zwracana wartość: *Integer*.

Parametr:

- **deviceUUID** - UUID urządzenia GPU.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.system.driver.version

 Wersja sterownika NVIDIA.
 Zwracana wartość: *String*.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

nvml.version

 Wersja biblioteki NVML.
 Zwracana wartość: *String*.

Więcej informacji można znaleźć w pliku readme wtyczki NVIDIA GPU.

oracle.diskgroups.stats[connString,<user>,<password>,<service>,<diskgroup>]

 Zwraca statystyki grup dyskowych Automatic Storage Management (ASM).
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **diskgroup** - nazwa grupy dyskowej ASM, dla której ma zostać wykonane zapytanie.

oracle.diskgroups.discovery[connString,<user>,<password>,<service>]

 Zwraca listę grup dyskowych ASM. Używane w odkrywaniu niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to będzie traktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; na końcu nie może występować spacja).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.archive.info[connString,<user>,<password>,<service>,<destination>]

 Statystyki logów archiwalnych.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od otwartego nawiasu ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **destination** - nazwa miejsca docelowego, które ma zostać odpytane.

oracle.cdb.info[connString,<user>,<password>,<service>,<database>]

 Informacje o bazach danych kontenerów (CDB).
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **database** - nazwa bazy danych, która ma zostać odpytywana.

oracle.custom.query[connString,<user>,<password>,<service>,queryName,<args...>]

 Wynik niestandardowego zapytania.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **queryName** - nazwa niestandardowego zapytania; musi odpowiadać nazwie pliku SQL bez rozszerzenia;
- **args** - jeden lub kilka argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

Uwagi: - Zwracane dane są automatycznie konwertowane do formatu JSON. - Należy unikać bezpośredniego zwracania JSON z zapytań, ponieważ ulegnie on uszkodzeniu, gdy wtyczka spróbuje ponownie przekonwertować go do formatu JSON.

oracle.datafiles.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki plików danych.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od otwartego nawiasu ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; na końcu nie może występować spacja).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.db.discovery[connString,<user>,<password>,<service>]

 Zwraca listę baz danych. Używane w odkrywaniu niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; na końcu nie może występować spacja).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.fra.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki Fast Recovery Area (FRA).
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; na końcu nie może występować spacja).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.instance.info[connString,<user>,<password>,<service>]

 Statystyki instancji.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;

    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego "(" (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; na końcu nie może występować spacja).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.pdb.info[connString,<user>,<password>,<service>,<database>]

 Informacje o bazach danych Pluggable Databases (PDBs).
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego "(" (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **database** - nazwa bazy danych, która ma zostać odpytana.

oracle.pdb.discovery[connString,<user>,<password>,<service>]

 Zwraca listę PDB. Używane w niskopoziomowym wykrywaniu.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego "(" (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania jest niewrażliwa na wielkość liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.pga.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki Program Global Area (PGA).
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. "tcp://"), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego "(" (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; na końcu nie może występować spacja).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.ping[connString,<user>,<password>,<service>]

 Sprawdza, czy można ustanowić połączenie z Oracle.
 Zwracana wartość: *1* - połączenie zostało nawiązane pomyślnie; *0* - połączenie jest zerwane (jeśli występuje jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

Przykłady:

```
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>]

 Zwraca statystyki procesów.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od otwartego nawiasu ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.redolog.info[connString,<user>,<password>,<service>]

 Informacje o pliku dziennika z pliku kontrolnego.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.sga.stats[connString,<user>,<password>,<service>]

 Zwraca statystyki System Global Area (SGA).
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.sessions.stats[connString,<user>,<password>,<service>,<lockMaxTime>]

 Zwraca statystyki sesji.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **lockMaxTime** - maksymalny czas trwania blokady sesji w sekundach, aby sesja została uznana za długotrwale zablokowaną. Wartość domyślna: 600 sekund.

oracle.sys.metrics[connString,<user>,<password>,<service>,<duration>]

 Zwraca zestaw wartości metryk systemowych.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a parametr `Plugins.Oracle.ResolveTNS` ma wartość false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i wartość zostanie potraktowana jako URI;
  - nazwa sesji - jeśli taka nazwa znajduje się w pliku konfiguracji wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS ma wartość true i nie zachodzi żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od otwartego nawiasu ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie uwzględnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **duration** - interwał przechwytywania (w sekundach) wartości metryk systemowych. Możliwe wartości: *60* - długi czas trwania (domyślnie), *15* - krótki czas trwania.

oracle.sys.params[connString,<user>,<password>,<service>]

 Zwraca zestaw wartości parametrów systemowych.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:

    - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.ts.stats[connString,<user>,<password>,<service>,<tablespace>,<type>,<conname>]

 Zwraca statystyki tablespace’ów.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **tablespace** - nazwa tablespace’a, dla którego ma zostać wykonane zapytanie. Wartość domyślna (jeśli pozostawiono puste i ustawiono `type`):
    - ”TEMP” (jeśli `type` jest ustawione na ”TEMPORARY”);
    - ”USERS” (jeśli `type` jest ustawione na ”PERMANENT”).
- **type** - typ tablespace’a, dla którego ma zostać wykonane zapytanie. Wartość domyślna (jeśli ustawiono `tablespace`): ”PERMANENT”.
- **conname** - nazwa kontenera, dla którego wymagane są informacje.

Jeśli pominięto `tablespace`, `type` lub `conname`, pozycja zwróci statystyki tablespace’ów dla wszystkich pasujących kontenerów (w tym PDB i CDB).

oracle.ts.discovery[connString,<user>,<password>,<service>]

 Zwraca listę tablespace’ów. Używane w niskopoziomowym wykrywaniu.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
    - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
    - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
    - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
    - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia; nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

oracle.user.info[connString,<user>,<password>,<service>,<username>]

 Zwraca informacje o użytkowniku Oracle.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba te elementy, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa została znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i żaden z powyższych warunków nie ma zastosowania;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (wielkość liter w opcji logowania nie ma znaczenia, nie może ona zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle;
- **username** - nazwa użytkownika, dla którego potrzebne są informacje. Nazwy użytkowników zapisane małymi literami nie są obsługiwane. Domyślnie: bieżący użytkownik.

oracle.version[connString,<user>,<password>,<service>]

 Zwraca wersję serwera bazy danych.
 Zwracana wartość: *String*.

Parametry:

- **connString** - może to być:
  - URI - jeśli nie znaleziono sesji o podanej nazwie, a `Plugins.Oracle.ResolveTNS` jest ustawione na false. Jeśli zawiera schemat (np. ”tcp://”), port (np. 1521) lub oba, opcja ResolveTNS nie będzie brana pod uwagę i mimo to zostanie potraktowane jako URI;
  - nazwa sesji - jeśli taka nazwa zostanie znaleziona w pliku konfiguracyjnym wtyczki;
  - klucz TNS - jeśli opcja ResolveTNS jest ustawiona na true i nie ma zastosowania żaden z powyższych warunków;
  - wartość TNS - jeśli zaczyna się od nawiasu otwierającego ”(” (początkowe spacje są ignorowane). W takim przypadku opcja ResolveTNS nie jest brana pod uwagę.
- **user** - nazwa użytkownika Oracle; obsługuje dołączenie jednej z opcji logowania `as sysdba`, `as sysoper`, `as sysasm`, `as sysbackup`, `as sysdg`, `as syskm` lub `as sysrac` w formacie `user as sysdba` (opcja logowania nie rozróżnia wielkości liter i nie może zawierać końcowej spacji).
- **password** - hasło Oracle;
- **service** - nazwa usługi Oracle.

pgsql.autovacuum.count[uri,<username>,<password>,<database name>]

 Liczba procesów autovacuum.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.archive[uri,<username>,<password>,<database name>]

 Informacje o zarchiwizowanych plikach.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.bgwriter[uri,<username>,<password>,<database name>]

 Łączna liczba punktów kontrolnych dla klastra bazy danych, z podziałem według typu punktu kontrolnego.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.cache.hit[uri,<username>,<password>,<database name>]

 Współczynnik trafień w buforze PostgreSQL.
 Wartość zwracana: *Float*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.connections[uri,<username>,<password>,<database name>]

 Zwraca połączenia według typu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.custom.query[uri,<username>,<password>,<database name>,queryName,<args...>]

 Zwraca wynik niestandardowego zapytania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość Plugins.PostgreSQL.Default.Database z pliku postgresql.conf);
- **queryName** - nazwa niestandardowego zapytania, musi odpowiadać nazwie pliku SQL bez rozszerzenia;
- **args** - jeden lub kilka argumentów rozdzielonych przecinkami, przekazywanych do zapytania.

pgsql.db.age[uri,<username>,<password>,<database name>]

 Wiek najstarszego FrozenXID bazy danych (liczba transakcji).
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.db.bloating_tables[uri,<username>,<password>,<database name>]

 Liczba bloating tables w każdej bazie danych.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.db.discovery[uri,<username>,<password>,<database name>]

 Lista baz danych PostgreSQL. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.db.size[uri,<username>,<password>,<database name>]

 Rozmiar bazy danych w bajtach.
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.dbstat[uri,<username>,<password>,<database name>]

 Zbiera statystyki dla każdej bazy danych. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.dbstat.sum[uri,<username>,<password>,<database name>]

 Zestawione dane dla wszystkich baz danych w klastrze.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.locks[uri,<username>,<password>,<database name>]

 Informacje o przyznanych blokadach w podziale na bazę danych. Używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.oldest.xid[uri,<username>,<password>,<database name>]

 Wiek najstarszego XID (liczba transakcji).
 Wartość zwracana: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.ping[uri,<username>,<password>,<database name>]

 Sprawdza, czy połączenie jest aktywne.
 Wartość zwracana: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.queries[uri,<username>,<password>,<database name>,time period]

 Zwraca metryki zapytań według czasu wykonania.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`);
- **time period** - limit czasu wykonania dla liczby wolnych zapytań (musi być dodatnią liczbą całkowitą).

pgsql.replication.count[uri,<username>,<password>]

 Liczba serwerów standby.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;

- **username, password** - poświadczenia PostgreSQL.

pgsql.replication.process[uri,<username>,<password>]

 Opóźnienie flush, opóźnienie zapisu i opóźnienie odtwarzania dla każdego procesu nadawcy.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.replication.process.discovery[uri,<username>,<password>]

 Odkrywanie nazw procesów replikacji.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL.

pgsql.replication.recovery_role[uri,<username>,<password>]

 Status odzyskiwania.
 Zwracana wartość: *0* - tryb master; *1* - odzyskiwanie jest nadal w toku (tryb standby).

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.replication.status[uri,<username>,<password>]

 Status replikacji.
 Zwracana wartość: *0* - streaming jest wyłączony; *1* - streaming jest włączony; *2* - tryb master.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.replication_lag.b[uri,<username>,<password>]

 Opóźnienie replikacji w bajtach.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL.

pgsql.replication_lag.sec[uri,<username>,<password>]

 Opóźnienie replikacji w sekundach.
 Zwracana wartość: *Integer*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL.

pgsql.uptime[uri,<username>,<password>,<database name>]

 Czas działania PostgreSQL w milisekundach.
 Zwracana wartość: *Float*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - dane uwierzytelniające PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięto, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.version[uri,<username>,<password>,<database name>]

 Zwraca wersję PostgreSQL.
 Wartość zwracana: *String*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

pgsql.wal.stat[uri,<username>,<password>,<database name>]

 Statystyki WAL.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **uri** - URI lub nazwa sesji;
- **username, password** - poświadczenia PostgreSQL;
- **database name** - nazwa bazy danych (jeśli pominięta, używana jest wartość `Plugins.PostgreSQL.Default.Database` z pliku `postgresql.conf`).

redis.config[connString,<password>,<pattern>]

 Pobiera parametry konfiguracyjne instancji Redis pasujące do wzorca.
 Zwracana wartość: *JSON* - jeśli użyto wzorca w stylu glob; pojedyncza wartość - jeśli wzorzec nie zawierał żadnego znaku wieloznacznego.

Parametry:

- **connString** - URI lub nazwa sesji;
- **password** - hasło Redis;
- **pattern** - wzorzec w stylu glob (domyślnie *).

Komentarz:

- Ostrzeżenie dotyczące bezpieczeństwa: `redis.config` uruchamia polecenie Redis `CONFIG GET` i może zwracać wrażliwe parametry konfiguracyjne (na przykład `requirepass`, `masterauth`, ustawienia związane z TLS oraz inne sekrety). Użycie domyślnego wzorca (`*`) lub szerokich wzorców może ujawnić te sekrety każdemu, kto ma uprawnienia do przeglądania wartości pozycji w Zabbix.

redis.info[connString,<password>,<section>]

 Pobiera dane wyjściowe polecenia INFO.
 Wartość zwracana: *JSON* — dane wyjściowe są serializowane do formatu JSON.

Parametry:

- **connString** — URI lub nazwa sesji;
- **password** — hasło Redis;
- **section** — sekcja informacji (domyślnie *default*).

redis.ping[connString,<password>]

 Sprawdza, czy połączenie jest aktywne, czy nie.
 Zwracana wartość: *1* - połączenie jest aktywne; *0* - połączenie jest zerwane (jeśli wystąpi jakikolwiek błąd, w tym problemy z AUTH i konfiguracją).

Parametry:

- **connString** - URI lub nazwa sesji;
- **password** - hasło Redis.

redis.slowlog.count[connString,<password>]

 Liczba wpisów w logu wolnych operacji od momentu uruchomienia Redis.
 Zwracana wartość: *Integer*.

Parametry:

- **connString** — URI lub nazwa sesji;
- **password** — hasło Redis.

smart.attribute.discovery

 Zwraca listę atrybutów urządzenia S.M.A.R.T.
 Wartość zwracana: *obiekt JSON*.

Uwagi:

- Zwracane są następujące makra i ich wartości: `{#NAME}`, `{#DISKTYPE}`, `{#ID}`, `{#ATTRNAME}`, `{#THRESH}`;
- Obsługiwane są typy dysków HDD, SSD i NVME. Dyski mogą być samodzielne lub połączone w RAID. `{#NAME}` będzie zawierać dodatek w przypadku RAID, np.: `{"{#NAME}": "/dev/sda cciss,2"}`.

smart.disk.discovery[<type>]

 Zwraca listę urządzeń S.M.A.R.T.
 Wartość zwracana: *obiekt JSON*.

Parametr:

- **type** - określa wartość używaną do skanowania dysków. Dostępne wartości: *id*, *name* (domyślnie). *id* nie jest obsługiwane w systemie Windows i po wybraniu zwróci błąd

Komentarze:

- Zwracane są następujące makra i ich wartości: {#NAME}, {#DISKTYPE}, {#MODEL}, {#SN}, {#PATH}, {#ATTRIBUTES}, {#RAIDTYPE};
- Obsługiwane są typy dysków HDD, SSD i NVME. Jeśli dysk nie należy do RAID, {#RAIDTYPE} będzie puste. {#NAME} będzie zawierać dodatek w przypadku RAID, np.: {"{#NAME}": "/dev/sda cciss,2"}.

smart.disk.get[<path>,<raid type>]

 Zwraca wszystkie dostępne właściwości urządzeń S.M.A.R.T.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **path** - ścieżka dysku, jako wartość można użyć makra {#PATH};
- **raid_type** - typ RAID, jako wartość można użyć makra {#RAID}

Komentarze:

- Obsługiwane są typy dysków HDD, SSD i NVME. Dyski mogą działać samodzielnie lub być połączone w RAID;
- Dane obejmują wersję smartctl i argumenty wywołania oraz dodatkowe pola:
disk_name - zawiera nazwę z wymaganymi dodatkami dla wykrywania RAID, np.: {"disk_name": "/dev/sda cciss,2"}
disk_type - zawiera typ dysku HDD, SSD lub NVME, np.: {"disk_type": "ssd"};
- Jeśli nie zostaną określone żadne parametry, pozycja zwróci informacje o wszystkich dyskach.

systemd.unit.get[unit name,<interface>]

 Zwraca wszystkie właściwości jednostki systemd.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **unit name** - nazwa jednostki (możesz użyć makra {#UNIT.NAME} w prototypie pozycji, aby wykryć nazwę);
- **interface** - typ interfejsu jednostki, możliwe wartości: *Unit* (domyślnie), *Service*, *Socket*, *Device*, *Mount*, *Automount*, *Swap*, *Target*, *Path*.

Uwagi:

- Ta pozycja jest obsługiwana tylko na platformie Linux;
- LoadState, ActiveState i UnitFileState dla interfejsu Unit są zwracane jako tekst i liczba całkowita: "ActiveState":{"state":1,"tex

systemd.unit.info[nazwa jednostki,<właściwość>,<interfejs>]

 Informacje o jednostce systemd.
 Zwracana wartość: *String*.

Parametry:

- **nazwa jednostki** - nazwa jednostki (możesz użyć makra {#UNIT.NAME} w prototypie pozycji, aby wykryć nazwę);
- **właściwość** - właściwość jednostki (np. ActiveState (domyślnie), LoadState, Description);
- **interfejs** - typ interfejsu jednostki (np. Unit (domyślnie), Socket, Service).

Uwagi:

- Ta pozycja jest obsługiwana tylko na platformie Linux;
- Ta pozycja umożliwia pobranie określonej właściwości określonego typu interfejsu zgodnie z opisem w dbus API.

Przykłady:

```
systemd.unit.info["{#UNIT.NAME}"] #zbieranie informacji o stanie aktywności (active, reloading, inactive,
systemd.unit.info["{#UNIT.NAME}",LoadState] #zbieranie informacji o stanie ładowania wykrytych jednostek s
systemd.unit.info[mysqld.service,Id] #pobieranie technicznej nazwy usługi (mysqld.service)
systemd.unit.info[mysqld.service,Description] #pobieranie opisu usługi (MySQL Server)
systemd.unit.info[mysqld.service,ActiveEnterTimestamp] #pobieranie ostatniego czasu przejścia usługi do st
systemd.unit.info[dbus.socket,NConnections,Socket] #zbieranie liczby połączeń z tej jednostki gniazda
```

systemd.unit.discovery[<type>]

 Lista jednostek systemd i ich szczegółów. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **type** — możliwe wartości: *all*, *automount*, *device*, *mount*, *path*, *service* (domyślnie), *socket*, *swap*, *target*.

Ta pozycja jest obsługiwana tylko na platformie Linux.

web.certificate.get[hostname,<port>,<address>]

 Weryfikuje certyfikaty i zwraca szczegóły certyfikatu.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **hostname** - może być adresem IP lub DNS.
Może zawierać schemat URL (*tylko https*), ścieżkę (zostanie zignorowana) oraz port.
Jeśli port zostanie podany zarówno w pierwszym, jak i w drugim parametrze, ich wartości muszą być zgodne.
Jeśli zostanie określony address (3. parametr), hostname jest używany tylko do SNI i weryfikacji hostname;
- **port** - numer portu (domyślnie 443 dla HTTPS);
- **address** - może być adresem IP lub DNS. Jeśli zostanie określony, będzie używany do połączenia, a hostname (1. parametr) będzie używany do SNI i weryfikacji host. W przypadku gdy 1. parametr jest adresem IP, a 3. parametr jest DNS, 1. parametr będzie używany do połączenia, a 3. parametr do SNI i weryfikacji host.

Komentarze:

- Ta pozycja staje się nieobsługiwana, jeśli cel określony w konfiguracji hosta nie istnieje, jest niedostępny lub jeśli uzgadnianie TLS zakończy się błędem innym niż nieprawidłowy certyfikat;
- Obecnie nie są obsługiwane rozszerzenie X.509 AIA (Authority Information Access), CRL i OCSP (w tym OCSP stapling) oraz Certificate Transparency;
- Pola odpowiedzi JSON:
  - *x509*: zawiera szczegóły certyfikatu X.509.
    - *version*: wersja X.509 (np. 3).
    - *serial_number*: numer seryjny certyfikatu.
    - *signature_algorithm*: algorytm użyty do podpisania certyfikatu (np. `SHA256-RSA`).
    - *issuer*: wystawca certyfikatu.
    - *not_before*: data rozpoczęcia ważności certyfikatu.
    - *not_after*: data wygaśnięcia certyfikatu.
    - *subject*: podmiot certyfikatu.
    - *public_key_algorithm*: algorytm użyty dla klucza publicznego (np. `RSA`).
    - *alternative_names*: alternatywne nazwy podmiotu (jeśli są obecne), w przeciwnym razie `null`.
  - result: zawiera wynik weryfikacji.
    - *value*: status weryfikacji (zobacz możliwe wartości poniżej).
    - *message*: szczegółowy komunikat weryfikacji (np. `"certificate verified successfully"`).
  - *sha1_fingerprint*: odcisk palca SHA-1 certyfikatu.
  - *sha256_fingerprint*: odcisk palca SHA-256 certyfikatu.
- Pole `$.result.value` wskazuje wynik weryfikacji certyfikatu. Możliwe wartości obejmują:
  - *valid* - certyfikat jest prawidłowy i zaufany.
  - *valid-but-self-signed* - certyfikat jest prawidłowy, ale samopodpisany, co oznacza, że jego podmiot jest zgodny z wystawcą.
  - *invalid* - certyfikat jest nieprawidłowy z powodu problemu, takiego jak wygaśnięcie, nieprawidłowy hostname lub nieznany urząd podpisujący.

Przykład:

```
web.certificate.get[example.com,443]
```

Odpowiedź 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"
}
```

Agent Zabbix dla systemu Windows

Przegląd

Pozycje agenta Zabbix dla systemu Windows są przedstawione na dwóch listach:

- Pozycje współdzielone — klucze pozycji współdzielone z agentem Zabbix dla systemów UNIX;
- Pozycje specyficzne dla Windows — klucze pozycji obsługiwane **tylko** w systemie Windows.

Należy pamiętać, że wszystkie klucze pozycji obsługiwane przez agent Zabbix w systemie Windows są również obsługiwane przez nowej generacji agent Zabbix 2. Zobacz dodatkowe klucze pozycji, których można używać wyłącznie z agentem 2.

Zobacz także: Minimalne uprawnienia dla pozycji Windows

Współdzielone pozycje

Poniższa tabela zawiera pozycje Zabbix agent, które są obsługiwane w systemie Windows i współdzielone z agentem UNIX Zabbix:

- Klucz pozycji jest łączem do pełnych szczegółów pozycji agenta UNIX Zabbix
- Uwzględniono komentarze dotyczące pozycji istotne dla systemu Windows

Item key	Description	Item group
log	Monitorowanie pliku dziennika. Ta pozycja nie jest obsługiwana dla Windows Event Log. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	Log monitoring
log.count	Liczba dopasowanych wierszy w monitorowanym pliku dziennika. Ta pozycja nie jest obsługiwana dla Windows Event Log. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	
logrt	Monitorowanie rotowanego pliku dziennika. Ta pozycja nie jest obsługiwana dla Windows Event Log. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	
logrt.count	Liczba dopasowanych wierszy w monitorowanym rotowanym pliku dziennika. Ta pozycja nie jest obsługiwana dla Windows Event Log. Parametr `persistent_dir` nie jest obsługiwany w systemie Windows.	
modbus.get	Odczytuje dane Modbus.	Modbus
net.dns	Sprawdza, czy usługa DNS działa. Parametry `ip`, `timeout` i `count` są ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2.	Network
net.dns.perf	Sprawdza wydajność usługi DNS. Parametry `ip`, `timeout` i `count` są ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2.	
net.dns.record	Wykonuje zapytanie DNS. Parametry `ip`, `timeout` i `count` są ignorowane w systemie Windows, chyba że używany jest Zabbix agent 2.	
net.if.discovery	Lista interfejsów sieciowych. Niektóre wersje systemu Windows (na przykład Server 2008) mogą wymagać zainstalowania najnowszych aktualizacji, aby obsługiwać znaki spoza ASCII w nazwach interfejsów.	
net.if.in	Statystyki ruchu przychodzącego na interfejsie sieciowym. W systemie Windows pozycja pobiera wartości z liczników 64-bitowych, jeśli są dostępne. Liczniki statystyk interfejsów 64-bitowych zostały wprowadzone w Windows Vista i Windows Server 2008. Jeśli liczniki 64-bitowe nie są dostępne, agent używa liczników 32-bitowych. Obsługiwane są wielobajtowe nazwy interfejsów w systemie Windows. Opisy interfejsów sieciowych w systemie Windows można uzyskać za pomocą pozycji net.if.discovery lub net.if.list.	

Item key	Description	Item group
net.if.out	Statystyki ruchu wychodzącego na interfejsie sieciowym. W systemie Windows pozycja pobiera wartości z liczników 64-bitowych, jeśli są dostępne. Liczniki statystyk interfejsów 64-bitowych zostały wprowadzone w Windows Vista i Windows Server 2008. Jeśli liczniki 64-bitowe nie są dostępne, agent używa liczników 32-bitowych. Obsługiwane są wielobajtowe nazwy interfejsów w systemie Windows. Opisy interfejsów sieciowych w systemie Windows można uzyskać za pomocą pozycji net.if.discovery lub net.if.list.	
net.if.total	Suma statystyk ruchu przychodzącego i wychodzącego na interfejsie sieciowym. W systemie Windows pozycja pobiera wartości z liczników 64-bitowych, jeśli są dostępne. Liczniki statystyk interfejsów 64-bitowych zostały wprowadzone w Windows Vista i Windows Server 2008. Jeśli liczniki 64-bitowe nie są dostępne, agent używa liczników 32-bitowych. Obsługiwane są wielobajtowe nazwy interfejsów w systemie Windows. Opisy interfejsów sieciowych w systemie Windows można uzyskać za pomocą pozycji net.if.discovery lub net.if.list.	
net.tcp.listen	Sprawdza, czy ten port TCP ma stan LISTEN.	
net.tcp.port	Sprawdza, czy można nawiązać połączenie TCP z określonym portem.	
net.tcp.service	Sprawdza, czy usługa działa i akceptuje połączenia TCP. Sprawdzanie LDAP i HTTPS w systemie Windows jest obsługiwane tylko przez Zabbix agent 2.	
net.tcp.service.perf	Sprawdza wydajność usługi TCP. Sprawdzanie LDAP i HTTPS w systemie Windows jest obsługiwane tylko przez Zabbix agent 2.	
net.tcp.socket.count	Zwraca liczbę gniazd TCP pasujących do parametrów. Ta pozycja jest obsługiwana w systemie Linux przez Zabbix agent, ale w systemie Windows jest obsługiwana tylko przez Zabbix agent 2 w 64-bitowym systemie Windows.	
net.udp.service	Sprawdza, czy usługa działa i odpowiada na żądania UDP.	
net.udp.service.perf	Sprawdza wydajność usługi UDP.	
net.udp.socket.count	Zwraca liczbę gniazd UDP pasujących do parametrów. Ta pozycja jest obsługiwana w systemie Linux przez Zabbix agent, ale w systemie Windows jest obsługiwana tylko przez Zabbix agent 2 w 64-bitowym systemie Windows.	
proc.get	Lista procesów systemu operacyjnego i ich parametrów. Parametr `cmdline` nie jest obsługiwany w systemie Windows.	Processes
proc.num	Liczba procesów. W systemie Windows obsługiwane są tylko parametry `name` i `user`.	
system.cpu.discovery	Lista wykrytych procesorów/rdzeni procesora.	System
system.cpu.load	Obciążenie CPU. Po uruchomieniu procesu kolektora na Zabbix agent inicjowane są następujące liczniki wydajności, które są później używane dla tej pozycji: `\System\Processor Queue Length`	
system.cpu.num	Liczba procesorów.	
system.cpu.util	Procent wykorzystania CPU. Wartość jest pobierana przy użyciu licznika wydajności *Processor Time*. Należy zauważyć, że od Windows 8 Menedżer zadań pokazuje wykorzystanie CPU na podstawie licznika wydajności *Processor Utility*, podczas gdy we wcześniejszych wersjach był to licznik *Processor Time* (zobacz więcej szczegółów). *system* jest jedynym parametrem `type` obsługiwanym w systemie Windows.	
system.hostname	Nazwa hosta systemu. Wartość jest pobierana w systemie Windows za pomocą funkcji `GetComputerName()` (dla **netbios**), `GetComputerNameExA()` (dla **fqdn**) lub `gethostname()` (dla **host**). Zobacz także bardziej szczegółowy opis.	
system.localtime	Czas systemowy.	
system.run	Uruchamia określone polecenie na host.	
system.sw.arch	Informacje o architekturze oprogramowania.	
system.sw.os	Informacje o systemie operacyjnym.	
system.sw.os.get	Szczegółowe informacje o systemie operacyjnym (wersja, typ, nazwa dystrybucji, wersja minor i major itp.).	

Item key	Description	Item group
system.swap.size	Rozmiar przestrzeni wymiany w bajtach lub jako procent całości. Parametr typu pused jest obsługiwany w systemie Linux przez Zabbix agent, ale w systemie Windows jest obsługiwany tylko przez Zabbix agent 2. Należy zauważyć, że ten klucz może zwracać nieprawidłowy rozmiar/procent przestrzeni wymiany na zwirtualizowanych platformach Windows (VMware ESXi, VirtualBox). W takim przypadku można użyć klucza `perf_counter[\700(_Total)\702]`, aby uzyskać poprawny procent przestrzeni wymiany.	
system.uname	Identyfikacja systemu. W systemie Windows wartość dla tej pozycji jest pobierana z klas WMI `Win32_OperatingSystem` i `Win32_Processor`. Nazwa systemu operacyjnego (w tym edycja) może być przetłumaczona na język wyświetlania użytkownika. W niektórych wersjach systemu Windows zawiera symbole znaku towarowego i dodatkowe spacje.	
system.uptime	Czas działania systemu w sekundach.	
vfs.dir.count	Liczba wpisów katalogu. W systemie Windows dowiązania symboliczne do katalogów są pomijane, a dowiązania twarde są liczone tylko raz.	Virtual file systems
vfs.dir.get	Lista wpisów katalogu. W systemie Windows dowiązania symboliczne do katalogów są pomijane, a dowiązania twarde są liczone tylko raz.	
vfs.dir.size	Rozmiar katalogu. W systemie Windows dowolne dowiązanie symboliczne jest pomijane, a dowiązania twarde są uwzględniane tylko raz.	
vfs.file.cksum	Suma kontrolna pliku, obliczana algorytmem UNIX cksum.	
vfs.file.contents	Pobieranie zawartości pliku.	
vfs.file.exists	Sprawdza, czy plik istnieje. W systemie Windows podwójne cudzysłowy muszą być poprzedzone znakiem ucieczki \\, a cały klucz pozycji musi być ujęty w podwójne cudzysłowy podczas korzystania z narzędzia wiersza poleceń do wywoływania `zabbix_get.exe` lub `agent2`. Należy zauważyć, że pozycja może stać się nieobsługiwana w systemie Windows, jeśli wyszukiwanie katalogu odbywa się w nieistniejącym katalogu, np. `vfs.file.exists[C:\no\dir,dir]` (gdzie 'no' nie istnieje).	
vfs.file.get	Zwraca informacje o pliku. Obsługiwane typy plików w systemie Windows: zwykły plik, katalog, dowiązanie symboliczne	
vfs.file.md5sum	Suma kontrolna MD5 pliku.	
vfs.file.owner	Pobiera właściciela pliku.	
vfs.file.regexp	Pobiera ciąg znaków z pliku.	
vfs.file.regmatch	Wyszukuje ciąg znaków w pliku.	
vfs.file.size	Rozmiar pliku.	
vfs.file.time	Informacje o czasie pliku. W systemie Windows XP `vfs.file.time[file,change]` może być równe `vfs.file.time[file,access]`.	
vfs.fs.discovery	Lista zamontowanych systemów plików wraz z ich typem i opcjami montowania. Makro `{#FSLABEL}` jest obsługiwane w systemie Windows.	
vfs.fs.get	Lista zamontowanych systemów plików wraz z ich typem, dostępnym miejscem na dysku, statystykami inode i opcjami montowania. Makro {#FSLABEL} jest obsługiwane w systemie Windows.	
vfs.fs.size	Miejsce na dysku w bajtach lub jako procent całości.	
vm.memory.size	Rozmiar pamięci w bajtach lub jako procent całości.	Virtual memory
web.page.get	Pobiera zawartość strony internetowej.	Web monitoring
web.page.perf	Czas ładowania pełnej strony internetowej.	
web.page.regexp	Wyszukuje ciąg znaków na stronie internetowej.	
agent.hostmetadata	Metadane hosta agenta.	Zabbix
agent.hostname	Nazwa hosta agenta.	
agent.ping	Sprawdzenie dostępności agenta.	
agent.variant	Wariant Zabbix agent (Zabbix agent lub Zabbix agent 2).	

Item key	Description	Item group
agent.version	Wersja Zabbix agent.	
zabbix.stats	Zwraca zdalnie zestaw wewnętrznych metryk serwera lub proxy Zabbix.	
zabbix.stats (queue)	Zwraca zdalnie liczbę monitorowanych pozycji w kolejce, które są opóźnione na serwerze Zabbix lub proxy.	

Pozycje specyficzne dla systemu Windows

Tabela zawiera szczegółowe informacje o kluczach pozycji, które są obsługiwane **wyłącznie** przez agenta Zabbix dla systemu Windows.

Pozycje specyficzne dla systemu Windows są czasami przybliżonym odpowiednikiem podobnej pozycji agenta, na przykład `proc_info`, obsługiwana w systemie Windows, w przybliżeniu odpowiada pozycji `proc.mem`, nieobsługiwanej w systemie Windows.

Klucz pozycji jest łączem do pełnych informacji o kluczu pozycji.

Klucz pozycji	Opis	Grupa pozycji
eventlog	Monitorowanie dziennika zdarzeń systemu Windows.	Monitorowanie logów
eventlog.count	Liczba wierszy w dzienniku zdarzeń systemu Windows.	
net.if.list	Lista interfejsów sieciowych (obejmuje typ interfejsu, stan, adres IPv4, opis).	Sieć
perf_counter	Wartość dowolnego licznika wydajności systemu Windows.	Liczniki wydajności
perf_counter_en	Wartość dowolnego licznika wydajności systemu Windows w języku angielskim.	
perf_instance.discovery	Lista instancji obiektów liczników wydajności systemu Windows.	
perf_instance_en.discovery	Lista instancji obiektów liczników wydajności systemu Windows, wykrytych przy użyciu nazw obiektów w języku angielskim.	
proc_info	Różne informacje o określonym procesie lub procesach.	Procesy
registry.data	Zwraca dane dla określonej nazwy wartości w kluczu Rejestru systemu Windows.	Rejestr
registry.get	Lista wartości lub kluczy Rejestru systemu Windows znajdujących się pod podanym kluczem.	
service.discovery	Lista usług systemu Windows.	Usługi
service.info	Informacje o usłudze.	
services	Wykaz usług.	
vm.vmemory.size	Rozmiar pamięci wirtualnej w bajtach lub jako wartość procentowa całkowitej pamięci.	Pamięć wirtualna
wmi.get	Wykonuje zapytanie WMI i zwraca pierwszy wybrany obiekt.	WMI
wmi.getall	Wykonuje zapytanie WMI i zwraca całą odpowiedź.	

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

eventlog[name,<regexp>,<severity>,<source>,<eventid>,<maxlines>,<mode>]

 Monitorowanie dziennika zdarzeń.
 Wartość zwracana: *Log*.

Parametry:

- **name** - nazwa kanału dziennika zdarzeń (*Log Name* w interfejsie GUI Podglądu zdarzeń);
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec (uwzględnia wielkość liter);
- **severity** - wyrażenie regularne opisujące poziom ważności (nie uwzględnia wielkości liter). Ten parametr akceptuje wyrażenie regularne oparte na następujących wartościach: ”Information”, ”Warning”, ”Error”, ”Critical”, ”Verbose” (działa w systemie Windows Vista lub nowszym).
- **source** - wyrażenie regularne opisujące identyfikator źródła (nie uwzględnia wielkości liter);
- **eventid** - wyrażenie regularne opisujące identyfikator(y) zdarzenia (uwzględnia wielkość liter);
- **maxlines** - maksymalna liczba nowych linii na sekundę, jaką agent wyśle do serwera Zabbix lub proxy. Ten parametr zastępuje wartość `MaxLinesPerSecond` w `zabbix_agentd.conf`.
- **mode** - możliwe wartości:
  - *all* (domyślnie);
  - *skip* - pomija przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).

Uwagi:

- Pozycja musi być skonfigurowana jako active check;
- Agent nie jest w stanie wysyłać zdarzeń z dziennika "Forwarded events";
- Obsługiwany jest Windows Eventing 6.0;
- Wybranie dla tej pozycji typu informacji innego niż Log spowoduje utratę lokalnego znacznika czasu, a także informacji o ważności dziennika i źródle;
- Zobacz także dodatkowe informacje o monitorowaniu dziennika.

Przykłady:

```
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] #tutaj odwołano się do niestandardowego wyrażenia regularnego o nazwie `TWOS
```

eventlog.count[name,<regexp>,<severity>,<source>,<eventid>,<maxproclines>,<mode>]

 Liczba wierszy w dzienniku zdarzeń systemu Windows.
 Wartość zwracana: *Integer*.

Parametry:

- **name** - nazwa kanału dziennika zdarzeń (*Log Name* w interfejsie GUI Podglądu zdarzeń);
- **regexp** - wyrażenie regularne expression opisujące wymagany wzorzec (uwzględnia wielkość liter);
- **severity** - wyrażenie regularne opisujące poziom ważności (nie uwzględnia wielkości liter). Ten parametr akceptuje wyrażenie regularne oparte na następujących wartościach: "Information", "Warning", "Error", "Critical", "Verbose" (uruchomione w systemie Windows Vista lub nowszym).
- **source** - wyrażenie regularne opisujące identyfikator źródła (nie uwzględnia wielkości liter);
- **eventid** - wyrażenie regularne opisujące identyfikator(y) zdarzenia (uwzględnia wielkość liter);
- **maxproclines** - maksymalna liczba nowych wierszy na sekundę, które agent będzie analizował (nie może przekroczyć 10000). Wartość domyślna to 10*MaxLinesPerSecond w `zabbix_agentd.conf`.
- **mode** - możliwe wartości:
  - *all* (domyślnie);
  - *skip* - pomiń przetwarzanie starszych danych (dotyczy tylko nowo utworzonych pozycji).

Uwagi:

- Pozycja musi być skonfigurowana jako active check;
- Agent nie jest w stanie wysyłać zdarzeń z dziennika "Forwarded events";
- Obsługiwany jest Windows Eventing 6.0;
- Wybranie dla tej pozycji innego niż Log type of information spowoduje utratę lokalnego znacznika czasu, a także informacji o ważności i źródle dziennika;
- Zobacz także dodatkowe informacje o monitorowaniu logów.

Przykłady:

```
eventlog.count[System,,"Warning|Error"]
eventlog.count[Windows PowerShell,,,,,,skip]
```

net.if.list

 Lista interfejsów sieciowych (zawiera typ interfejsu, status, adres IPv4, opis).
 Zwracana wartość: *Tekst*.

Uwagi:

- Obsługiwane są nazwy interfejsów wielobajtowych;
- Wyłączone interfejsy nie są wyświetlane;
- Włączanie/wyłączanie niektórych komponentów może zmienić ich kolejność w nazwie interfejsu systemu Windows;
- Niektóre wersje systemu Windows (na przykład Server 2008) mogą wymagać zainstalowania najnowszych aktualizacji, aby obsługiwać znaki spoza ASCII w nazwach interfejsów.

perf_counter[counter,<interval>]

 Wartość dowolnego licznika wydajności systemu Windows.
 Wartość zwracana: *Integer*, *float*, *string* lub *text* (w zależności od zapytania).

Parametry:

- **counter** - ścieżka do licznika;

- **interval** - ostatnie N sekund do przechowywania wartości średniej. `interval` musi mieścić się w zakresie od 1 do 900 sekund (włącznie), a wartość domyślna to 1.

Komentarze:

- Ta pozycja obsługuje ścieżki UNC, co może stanowić zagrożenie bezpieczeństwa;
- `interval` jest używany dla liczników wymagających więcej niż jednej próbki (takich jak wykorzystanie CPU), więc sprawdzenie za każdym razem zwraca średnią wartość z ostatnich ”interval” sekund;
- Program Performance Monitor może być użyty do uzyskania listy dostępnych liczników.
- Zobacz także: Windows performance counters.

perf_counter_en[counter,<interval>]

 Wartość dowolnego licznika wydajności systemu Windows w języku angielskim.
 Wartość zwracana: *Integer*, *float*, *string* lub *text* (w zależności od żądania).

Parametry:

- **counter** - ścieżka do licznika w języku angielskim;
- **interval** - ostatnie N sekund do przechowywania wartości średniej. `interval` musi mieścić się w zakresie od 1 do 900 sekund (włącznie), a wartość domyślna to 1.

Uwagi:

- Ta pozycja korzysta ze ścieżek UNC, co może stanowić zagrożenie bezpieczeństwa;
- `interval` jest używany dla liczników wymagających więcej niż jednej próbki (takich jak użycie CPU), więc sprawdzenie za każdym razem zwraca wartość średnią z ostatnich ”interval” sekund;
- Ta pozycja jest obsługiwana tylko w **Windows Server 2008/Vista** i nowszych;
- Listę angielskich ciągów można znaleźć, przeglądając następujący klucz rejestru: `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Wir NT\CurrentVersion\Perflib\009`.

perf_instance.discovery[object]

 Lista instancji obiektów liczników wydajności systemu Windows. Używana do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **object** — nazwa obiektu (zlokalizowana).

perf_instance_en.discovery[object]

 Lista instancji obiektów liczników wydajności systemu Windows, wykrywanych przy użyciu nazw obiektów w języku angielskim. Używane do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

Parametr:

- **object** - nazwa obiektu (w języku angielskim).

proc_info[process,<attribute>,<type>]

 Różne informacje o określonym procesie lub procesach.
 Wartość zwracana: *Float*.

Parametry:

- **process** - nazwa procesu;
- **attribute** - żądany atrybut procesu;
- **type** - typ reprezentacji (ma znaczenie, gdy istnieje więcej niż jeden proces o tej samej nazwie)

Komentarze:

- Obsługiwane są następujące `attributes`:
  - *vmsize* (domyślnie) - rozmiar wirtualnej pamięci procesu w Kbytes;
  - *wkset* - rozmiar working set procesu (ilość pamięci fizycznej używanej przez proces) w Kbytes;
  - *pf* - liczba page faults;
  - *ktime* - czas jądra procesu w milisekundach;
  - *utime* - czas użytkownika procesu w milisekundach;
  - *io_read_b* - liczba bajtów odczytanych przez proces podczas operacji I/O;
  - *io_read_op* - liczba operacji odczytu wykonanych przez proces;
  - *io_write_b* - liczba bajtów zapisanych przez proces podczas operacji I/O;
  - *io_write_op* - liczba operacji zapisu wykonanych przez proces;
  - *io_other_b* - liczba bajtów przesłanych przez proces podczas operacji innych niż odczyt i zapis;
  - *io_other_op* - liczba operacji I/O wykonanych przez proces, innych niż odczyt i zapis;
  - *gdiobj* - liczba obiektów GDI używanych przez proces;

    - *userobj* - liczba obiektów USER używanych przez proces.
- Prawidłowe `types` to:
    - *avg* (domyślnie) - średnia wartość dla wszystkich procesów o nazwie <process>;
    - *min* - minimalna wartość spośród wszystkich procesów o nazwie <process>;
    - *max* - maksymalna wartość spośród wszystkich procesów o nazwie <process>;
    - *sum* - suma wartości dla wszystkich procesów o nazwie <process>.
- W systemie 64-bitowym do poprawnego działania tej pozycji wymagany jest 64-bitowy agent Zabbix.

Przykłady:

```
proc_info[iexplore.exe,wkset,sum] #pobranie ilości pamięci fizycznej zajmowanej przez wszystkie procesy In
proc_info[iexplore.exe,pf,avg] #pobranie średniej liczby page faults dla procesów Internet Explorer
```

registry.data[key,<value name>]

 Zwraca dane dla określonej nazwy wartości w kluczu rejestru Windows.
 Zwracana wartość: *Integer*, *string* lub *text* (w zależności od typu wartości)

Parametry:

- **key** - klucz rejestru, łącznie z kluczem głównym; dozwolone są skróty kluczy głównych (np. HKLM);
- **value name** - nazwa wartości rejestru w kluczu (domyślnie pusty ciąg znaków ””). Jeśli nazwa wartości nie zostanie podana, zwracana jest wartość domyślna.

Uwagi:

- Obsługiwane skróty kluczy głównych:
    - 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
- Klucze zawierające spacje muszą być ujęte w podwójne cudzysłowy.

Przykłady:

```
registry.data["HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting"] #zwraca dane wartoś
registry.data["HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting","EnableZip"] #zwraca dane wartości
```

registry.get[key,<mode>,<name regexp>]

 Lista wartości lub kluczy rejestru Windows znajdujących się pod podanym kluczem.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **key** - klucz rejestru wraz z kluczem głównym; dozwolone są skróty kluczy głównych (np. HKLM) (pełną listę skrótów można znaleźć w komentarzach do `registry.data[]`);
- **mode** - możliwe wartości:
values (domyślnie) lub *keys*;
- **name regexp** - wykrywa tylko wartości, których nazwy pasują do wyrażenia regularnego (domyślnie - wykrywane są wszystkie wartości). Dozwolone tylko przy *values* jako *mode*.

Klucze zawierające spacje muszą być ujęte w podwójne cudzysłowy.

Przykłady:

```
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values,"^DisplayName|DisplayVersion$
JSON będzie zawierał szczegóły klucza, ostatniego podklucza, nazwy wartości, typu wartości i danych wartoś
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,values] #zwraca dane wszystkich wart
JSON będzie zawierał szczegóły klucza, ostatniego podklucza, nazwy wartości, typu wartości i danych wartoś
registry.get[HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,keys] #zwraca wszystkie podklucze te
JSON będzie zawierał szczegóły klucza i ostatniego podklucza.
```

service.discovery

 Lista usług Windows. Używana do odkrywania niskiego poziomu.
 Zwracana wartość: *obiekt JSON*.

service.info[service,<param>]

 Informacje o usłudze.
 Wartość zwracana: - *Integer* - z `param` jako *state*, *startup*;

 W szczególności dla *state*: - 0 - uruchomiona - 1 - wstrzymana - 2 - oczekiwanie na uruchomienie - 3 - oczekiwanie na wstrzymanie - 4 - oczekiwanie na wznowienie - 5 - oczekiwanie na zatrzymanie - 6 - zatrzymana - 7 - nieznana - 255 - brak takiej usługi

 W szczególności dla *startup*: - 0 - automatyczny - 1 - automatyczny opóźniony - 2 - ręczny - 3 - wyłączony - 4 - nieznany - 5 - automatyczne uruchamianie przez wyzwalacz - 6 - automatyczne opóźnione uruchamianie przez wyzwalacz - 7 - ręczne uruchamianie przez wyzwalacz - *String* - z `param` jako *displayname*, *path*, *user*; - *Text* - z `param` jako *description*.

Parametry:

- **service** - rzeczywista nazwa usługi lub jej nazwa wyświetlana widoczna w przystawce Usługi MMC;
- **param** - *state* (domyślnie), *displayname*, *path*, *user*, *startup* lub *description*.

Uwagi:

- Pozycje takie jak `service.info[service,state]` i `service.info[service]` zwrócą te same informacje;
- Tylko przy `param` ustawionym na *state* ta pozycja zwraca wartość dla nieistniejących usług (255).

Przykłady:

```
service.info[SNMPTRAP] - stan usługi SNMPTRAP;
service.info[SNMP Trap] - stan tej samej usługi, ale z podaną nazwą wyświetlaną;
service.info[EventLog,startup] - typ uruchamiania usługi EventLog
```

services[<type>,<state>,<exclude>]

 Lista usług.
 Zwracana wartość: *0* - jeśli pusta; *Text* - lista usług rozdzielonych znakiem nowej linii.

Parametry:

- **type** - *all* (domyślnie), *automatic*, *manual* lub *disabled*;
- **state** - *all* (domyślnie), *stopped*, *started*, *start_pending*, *stop_pending*, *running*, *continue_pending*, *pause_pending* lub *paused*;
- **exclude** - usługi, które należy wykluczyć z wyniku. Wykluczane usługi powinny być podane w podwójnych cudzysłowach, rozdzielone przecinkami, bez spacji.

Przykłady:

```
services[,started] #zwraca listę uruchomionych usług;
services[automatic, stopped] #zwraca listę zatrzymanych usług, które powinny być uruchomione;
services[automatic, stopped, "service1,service2,service3"] #zwraca listę zatrzymanych usług, które powinny
```

vm.vmemory.size[<type>]

 Rozmiar pamięci wirtualnej w bajtach lub jako procent całości.
 Zwracana wartość: *Integer* - dla bajtów; *float* - dla wartości procentowej.

Parametr:

- **type** - możliwe wartości: *available* (dostępna pamięć wirtualna), *pavailable* (dostępna pamięć wirtualna, w procentach), *pused* (używana pamięć wirtualna, w procentach), *total* (całkowita pamięć wirtualna, domyślnie) lub *used* (używana pamięć wirtualna)

Komentarze:

- Monitorowanie statystyk pamięci wirtualnej opiera się na:
  - Całkowitej pamięci wirtualnej w systemie Windows (całkowita pamięć fizyczna + rozmiar pliku stronicowania);
  - Maksymalnej ilości pamięci, którą agent Zabbix może przydzielić;
  - Bieżącym limicie pamięci przydzielonej dla systemu lub agent Zabbix, w zależności od tego, która wartość jest mniejsza.

Przykład:

```
vm.vmemory.size[pavailable] #zwraca dostępną pamięć wirtualną, w procentach
```

wmi.get[<namespace>,<query>]

 Wykonuje zapytanie WMI i zwraca pierwszy wybrany obiekt.
 Wartość zwracana: *Integer*, *float*, *string* lub *text* (w zależności od żądania).

Parametry:

- **namespace** - przestrzeń nazw WMI;
- **query** - zapytanie WMI zwracające pojedynczy obiekt.

Zapytania WMI są wykonywane przy użyciu WQL.

Przykład:

```
wmi.get[root\cimv2,select status from Win32_DiskDrive where Name like '%PHYSICALDRIVE0%'] #zwraca status p
```

wmi.getall[<namespace>,<query>]

 Wykonuje zapytanie WMI i zwraca całą odpowiedź. Może być używane do odkrywania niskiego poziomu.
 Wartość zwracana: *obiekt JSON*

Parametry:

- **namespace** - przestrzeń nazw WMI;
- **query** - zapytanie WMI.

Uwagi:

- Zapytania WMI są wykonywane przy użyciu WQL.
- Przetwarzanie wstępne JSONPath może być używane do wskazania bardziej szczegółowych wartości w zwróconym JSON.

Przykład:

```
wmi.getall[root\cimv2,select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'] #zwraca informacje
```

Monitorowanie usług Windows

Ten samouczek zawiera instrukcje krok po kroku dotyczące konfiguracji monitorowania usług Windows. Zakłada się, że serwer Zabbix i agent są skonfigurowane i działają poprawnie.

Krok 1

Uzyskaj nazwę usługi.

Nazwę usługi można uzyskać, przechodząc do przystawki MMC Usługi i otwierając właściwości usługi. Na karcie *Ogólne* powinno być widoczne pole o nazwie *Nazwa usługi*. Wartość, która po nim następuje, to nazwa, której użyjesz podczas konfigurowania pozycji do monitorowania. Na przykład, jeśli chcesz monitorować usługę "workstation", nazwa usługi może brzmieć: **lanmanworkstation**.

Krok 2

Skonfiguruj pozycję do monitorowania usługi.

Pozycja `service.info[service,<param>]` pobiera informacje o określonej usłudze. W zależności od potrzebnych informacji należy określić opcję `param`, która przyjmuje następujące wartości: *displayname*, *state*, *path*, *user*, *startup* lub *description*. Wartością domyślną jest *state*, jeśli `param` nie zostanie określony (`service.info[service]`).

Typ zwracanej wartości zależy od wybranego `param`: integer dla *state* i *startup*; character string dla *displayname*, *path* i *user*; text dla *description*.

Przykład:

- *Klucz:* `service.info[lanmanworkstation]`
- *Typ informacji:* Numeryczny (bez znaku)

Pozycja `service.info[lanmanworkstation]` pobierze informacje o stanie usługi jako wartość liczbową. Aby odwzorować wartość liczbową na reprezentację tekstową we frontendzie („0" jako „Running", „1" jako „Paused" itd.), można skonfigurować mapowanie wartości na hoście, na którym skonfigurowano pozycję. Aby to zrobić, należy albo podłączyć szablon *Windows services by Zabbix agent* lub *Windows services by Zabbix agent active* do hosta, albo skonfigurować na hoście nową mapę wartości opartą na mapie wartości *Windows service state* skonfigurowanej we wspomnianych szablonach.

Należy pamiętać, że oba wspomniane szablony mają skonfigurowaną regułę wykrywania, która będzie automatycznie wykrywać usługi. Jeśli nie jest to pożądane, można wyłączyć regułę wykrywania na poziomie hosta po podłączeniu szablonu do hosta.

Wykrywanie usług Windows

Wykrywanie niskiego poziomu zapewnia sposób na automatyczne tworzenie pozycji, wyzwalaczy i wykresów dla różnych obiektów na komputerze. Zabbix może automatycznie rozpocząć monitorowanie usług Windows na komputerze, bez potrzeby znajomości dokładnej nazwy usługi ani ręcznego tworzenia pozycji dla każdej usługi. Filtr może zostać użyty do generowania rzeczywistych pozycji, wyzwalaczy i wykresów tylko dla interesujących usług.

Monitorowanie pliku dziennika

Przegląd

Zabbix może być używany do scentralizowanego monitorowania i analizy plików dziennika z obsługą lub bez obsługi rotacji logów.

Powiadomienia mogą być używane do ostrzegania użytkowników, gdy plik dziennika zawiera określone ciągi znaków lub wzorce ciągów.

Aby monitorować plik dziennika, musisz mieć:

- Uruchomiony agent Zabbix na hoście.
- Skonfigurowaną pozycję monitorowania logów.

Attention:
Limit rozmiaru monitorowanego pliku dziennika zależy od obsługi dużych plików.

Konfiguracja

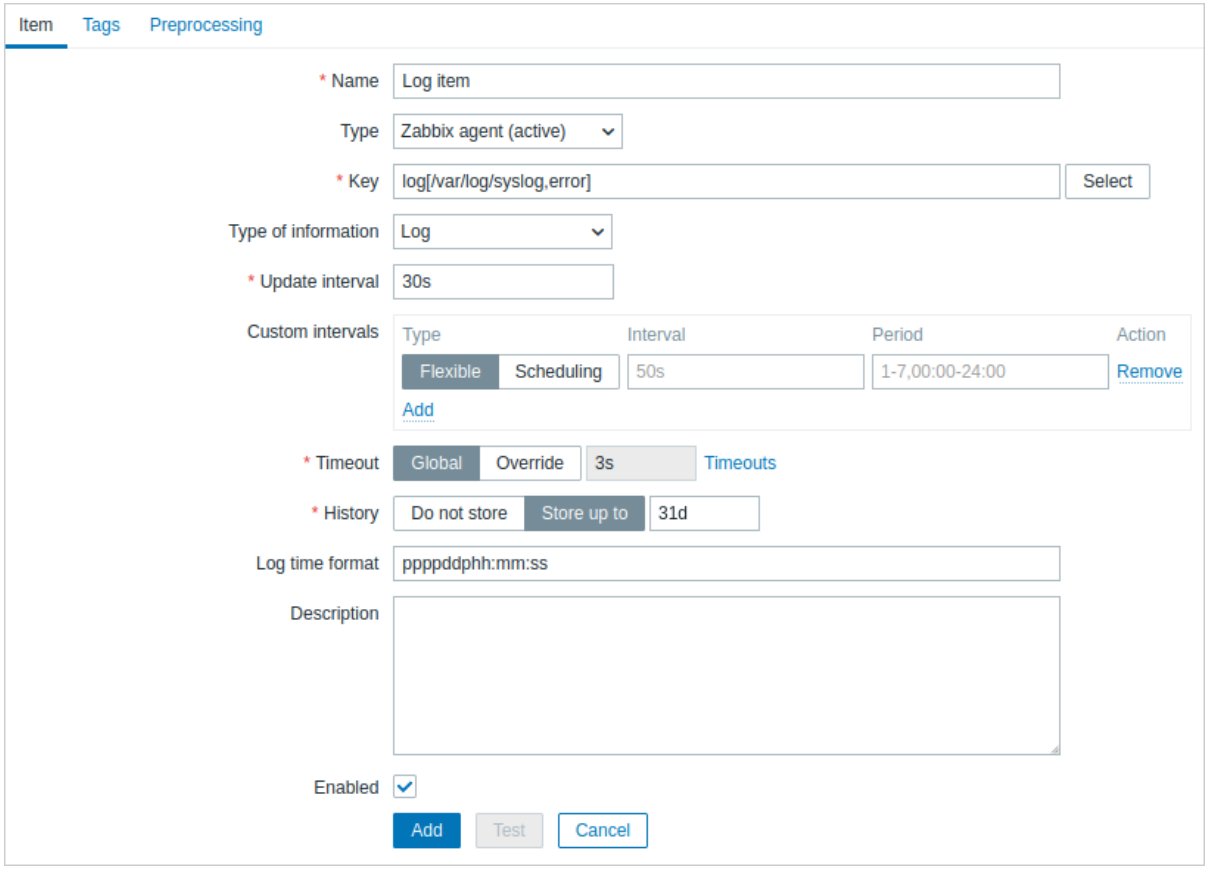
Zweryfikuj parametry agenta

Upewnij się, że w pliku konfiguracyjnym agenta:

- parametr `Hostname` odpowiada nazwie hosta w frontend.
- serwery w parametrze `ServerActive` są określone do przetwarzania aktywnych kontroli.

Konfiguracja pozycji

Skonfiguruj pozycję monitorowania dzienników.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

W przypadku pozycji monitorowania dzienników należy wprowadzić:

Typ	Wybierz tutaj **Zabbix agent (active)**.

Klucz	Użyj jednego z następujących kluczy pozycji: `log[]` lub `logrt[]`: Te dwa klucze pozycji umożliwiają monitorowanie dzienników i filtrowanie wpisów dziennika według wyrażenia regularnego zawartości, jeśli zostało podane. Na przykład: `log[/var/log/syslog,error]`. Upewnij się, że użytkownik „zabbix" ma uprawnienia do odczytu pliku; w przeciwnym razie status pozycji zostanie ustawiony na „unsupported". **log.count[]** lub **logrt.count[]**: Te dwa klucze pozycji umożliwiają zwracanie wyłącznie liczby pasujących wierszy. Szczegółowe informacje dotyczące używania tych kluczy pozycji i ich parametrów zawiera sekcja dotycząca obsługiwanych kluczy pozycji Zabbix agent.
Typ informacji	Wstępnie wypełniany automatycznie: Dla pozycji `log[]` lub `logrt[]` - *Log*; Dla pozycji `log.count[]` lub `logrt.count[]` - *Numeric (unsigned)*. Jeśli opcjonalnie używasz parametru `output`, możesz ręcznie wybrać odpowiedni typ informacji inny niż *Log*. Pamiętaj, że wybranie typu informacji innego niż Log spowoduje utratę lokalnego znacznika czasu.
Interwał aktualizacji (w sekundach)	Parametr określa, jak często agent Zabbix będzie sprawdzać, czy w pliku dziennika zaszły zmiany. Ustawienie wartości 1 sekundy zapewni otrzymywanie nowych rekordów tak szybko, jak to możliwe.
Format czasu dziennika	W tym polu można opcjonalnie określić wzorzec analizowania znacznika czasu w wierszu dziennika. Obsługiwane symbole zastępcze: * **y**: *Rok (1970-2038)* * **M**: *Miesiąc (01-12)* * **d**: *Dzień (01-31)* * **h**: *Godzina (00-23)* * **m**: *Minuta (00-59)* * **s**: *Sekunda (00-59)* Jeśli pole pozostanie puste, znacznik czasu zostanie ustawiony na 0 w czasie uniksowym, co oznacza `January 1, 1970`. Rozważmy na przykład następujący wiersz z pliku dziennika agenta Zabbix: `23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211).` Rozpoczyna się on sześcioma znakami pozycji PID, po których następują data, godzina i pozostała część komunikatu. Format czasu dziennika dla tego wiersza to „pppppp:yyyyMMdd:hhmmss". Pamiętaj, że znaki „p" i „:" są symbolami zastępczymi i mogą być dowolnymi znakami z wyjątkiem „yMdhms".

Ważne uwagi

- Serwer i agent przechowują ślad rozmiaru monitorowanego logu oraz czasu ostatniej modyfikacji (dla `logrt`) w dwóch licznikach. Dodatkowo:
  - Agent wewnętrznie używa także numerów inode (w UNIX/GNU/Linux), indeksów plików (w Microsoft Windows) oraz sum MD5 pierwszych 512 bajtów pliku logu, aby poprawić decyzje w sytuacjach, gdy pliki logów są obcinane i rotowane.
  - W systemach UNIX/GNU/Linux zakłada się, że systemy plików, na których są przechowywane pliki logów, zwracają numery inode, które można wykorzystać do śledzenia plików.
  - W systemie Microsoft Windows agent Zabbix określa typ systemu plików, na którym znajdują się pliki logów, i używa:
    * W systemach plików NTFS 64-bitowych indeksów plików.
    * W systemach plików ReFS (tylko od Microsoft Windows Server 2012) 128-bitowych identyfikatorów plików.
    * W systemach plików, w których indeksy plików się zmieniają (np. FAT32, exFAT), używany jest algorytm awaryjny, aby przyjąć rozsądne podejście w niepewnych warunkach, gdy rotacja pliku logu powoduje wiele plików logów z tym samym czasem ostatniej modyfikacji.
  - Numery inode, indeksy plików i sumy MD5 są wewnętrznie zbierane przez agenta Zabbix. Nie są przesyłane do serwera Zabbix i są tracone po zatrzymaniu agenta Zabbix.
  - Nie należy modyfikować czasu ostatniej modyfikacji pliku logu (na przykład za pomocą `touch`) ani zastępować monitorowanego pliku logu przez skopiowanie pliku z powrotem pod jego oryginalną nazwę (tworzy to nowy inode). W obu przypadkach Zabbix może potraktować plik jako inny plik i odczytać go ponownie od początku, co może spowodować duplikaty alertów.
  - Jeśli dla pozycji `logrt[]` istnieje kilka pasujących plików logów, a agent Zabbix śledzi najnowszy z nich i ten najnowszy plik logu zostanie usunięty, w dzienniku zostanie zapisany komunikat ostrzegawczy "`there are no files`

matching "<regexp mask>" in "<directory>". Agent Zabbix ignoruje pliki logów z czasem modyfikacji wcześniejszym niż najnowszy czas modyfikacji widziany przez agenta dla sprawdzanej pozycji logrt[].

- Agent rozpoczyna odczyt pliku logu od miejsca, w którym zakończył poprzednio.
- Liczba bajtów już przeanalizowanych (licznik rozmiaru) oraz czas ostatniej modyfikacji (licznik czasu) są przechowywane w bazie danych Zabbix i są wysyłane do agenta, aby upewnić się, że agent rozpocznie odczyt pliku logu od tego miejsca w przypadkach, gdy agent został właśnie uruchomiony lub otrzymał pozycje, które wcześniej były wyłączone albo nie były obsługiwane. Jednak jeśli agent otrzymał od serwera niezerowy licznik rozmiaru, ale pozycja logrt[] lub logrt.count[] nie może znaleźć pasujących plików, licznik rozmiaru jest resetowany do 0, aby analizować od początku, jeśli pliki pojawią się później.
- Za każdym razem, gdy plik logu stanie się mniejszy niż znany agentowi licznik rozmiaru logu, licznik jest resetowany do zera, a agent rozpoczyna odczyt pliku logu od początku, biorąc pod uwagę licznik czasu.
- Jeśli w katalogu znajduje się kilka pasujących plików o tym samym czasie ostatniej modyfikacji, agent próbuje poprawnie przeanalizować wszystkie pliki logów z tym samym czasem modyfikacji i uniknąć pominięcia danych lub ich podwójnej analizy, choć nie można tego zagwarantować we wszystkich sytuacjach. Agent nie zakłada żadnego konkretnego schematu rotacji plików logów ani go nie określa. W przypadku wielu plików logów z tym samym czasem ostatniej modyfikacji agent przetwarza je w kolejności leksykograficznie malejącej. W związku z tym dla niektórych schematów rotacji pliki logów będą analizowane i raportowane w ich oryginalnej kolejności. Dla innych schematów rotacji oryginalna kolejność plików logów nie będzie zachowana, co może prowadzić do raportowania dopasowanych rekordów pliku logu w zmienionej kolejności (problem nie występuje, jeśli pliki logów mają różne czasy ostatniej modyfikacji).
- Agent Zabbix przetwarza nowe rekordy pliku logu raz na *Update interval* sekund.
- Agent Zabbix nie wysyła więcej niż maxlines pliku logu na sekundę. Limit ten zapobiega przeciążeniu zasobów sieci i CPU oraz zastępuje wartość domyślną podaną przez parametr MaxLinesPerSecond w pliku konfiguracyjnym agenta.
- Aby znaleźć wymagany ciąg, Zabbix przetworzy 10 razy więcej nowych linii niż ustawiono w MaxLinesPerSecond. Tak więc na przykład, jeśli pozycja log[] lub logrt[] ma *Update interval* równy 1 sekundzie, domyślnie agent przeanalizuje nie więcej niż 200 rekordów pliku logu i wyśle do serwera Zabbix nie więcej niż 20 pasujących rekordów w jednym sprawdzeniu. Zwiększając MaxLinesPerSecond w pliku konfiguracyjnym agenta lub ustawiając parametr maxlines w kluczu pozycji, można zwiększyć limit do 10000 analizowanych rekordów pliku logu i 1000 pasujących rekordów wysyłanych do serwera Zabbix w jednym sprawdzeniu. Jeśli *Update interval* jest ustawiony na 2 sekundy, limity dla jednego sprawdzenia będą 2 razy wyższe niż przy *Update interval* równym 1 sekundzie.
- Dodatkowo wartości log i log.count są zawsze ograniczone do 50% rozmiaru bufora wysyłania agenta, nawet jeśli nie ma w nim żadnych wartości innych niż log. Aby więc wartości maxlines mogły zostać wysłane w jednym połączeniu (a nie w kilku), parametr agenta BufferSize musi wynosić co najmniej maxlines x 2. Agent Zabbix może przesyłać dane podczas zbierania logów i dzięki temu zwalniać bufor, natomiast agent Zabbix 2 wstrzyma zbieranie logów do czasu przesłania danych i zwolnienia bufora, co odbywa się asynchronicznie.
- W przypadku braku pozycji logów cały rozmiar bufora agenta jest używany dla wartości innych niż log. Gdy pojawią się wartości logów, zastępują one starsze wartości inne niż log, w razie potrzeby, do wyznaczonych 50%.
- Dla rekordów pliku logu dłuższych niż 256kB tylko pierwsze 256kB są dopasowywane do wyrażenia regularnego, a reszta rekordu jest ignorowana. Jednak jeśli agent Zabbix zostanie zatrzymany podczas przetwarzania długiego rekordu, wewnętrzny stan agenta zostanie utracony i długi rekord może zostać przeanalizowany ponownie i inaczej po ponownym uruchomieniu agenta.
- Szczególna uwaga dotycząca separatorów ścieżek \: jeśli file_format to file\.log, to nie powinien istnieć katalog file, ponieważ nie da się jednoznacznie określić, czy . jest znakiem escape, czy pierwszym symbolem nazwy pliku.
- Wyrażenia regularne dla logrt są obsługiwane tylko w nazwie pliku, dopasowywanie wyrażenia regularnego dla katalogu nie jest obsługiwane.
- Na platformach UNIX pozycja logrt[] staje się NOTSUPPORTED, jeśli katalog, w którym oczekuje się plików logów, nie istnieje.
- W systemie Microsoft Windows, jeśli katalog nie istnieje, pozycja nie stanie się NOTSUPPORTED (na przykład, jeśli nazwa katalogu jest błędnie wpisana w kluczu pozycji).
- Brak plików logów dla pozycji logrt[] nie powoduje stanu NOTSUPPORTED. Błędy odczytu plików logów dla pozycji logrt[] są zapisywane jako ostrzeżenia w pliku logu agenta Zabbix, ale nie powodują stanu NOTSUPPORTED.
- Plik logu agenta Zabbix może być pomocny w ustaleniu, dlaczego pozycja log[] lub logrt[] stała się NOTSUPPORTED. Zabbix może monitorować własny plik logu agenta, z wyjątkiem poziomów DebugLevel=4 lub DebugLevel=5.
- Wyszukiwanie znaku zapytania za pomocą wyrażenia regularnego, np. \?, może dawać wyniki fałszywie dodatnie, jeśli plik tekstowy zawiera symbole NUL, ponieważ są one zastępowane przez ? przez Zabbix, aby kontynuować przetwarzanie linii aż do znaku nowej linii.

Wyodrębnianie pasującego fragmentu wyrażenia regularnego

Czasami chcemy wyodrębnić tylko interesującą nas wartość z pliku docelowego zamiast zwracać cały wiersz po znalezieniu dopasowania wyrażenia regularnego.

Pozycje typu log mogą wyodrębniać żądane wartości z dopasowanych wierszy. Umożliwia to dodatkowy parametr **output** w pozycjach log i logrt.

Użycie parametru „output” pozwala wskazać „grupę przechwytującą” dopasowania, która może nas interesować.

Na przykład:

```
log[/path/to/the/file,"large result buffer allocation.*Entries: ([0-9]+)",,,,\1]
```

powinno umożliwić zwrócenie liczby wpisów znalezionej w treści:

```
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
```

Zostanie zwrócona tylko liczba, ponieważ \1 odnosi się do jedynej i pierwszej grupy przechwytującej: ([0-9]+).

Dzięki możliwości wyodrębniania i zwracania liczby wartość może być używana do definiowania wyzwalaczy.

Używanie parametru maxdelay

Parametr maxdelay w pozycjach logów pozwala ignorować niektóre starsze wiersze z plików logów, aby w ramach maxdelay sekund analizowane były najnowsze wiersze.

> **Warning:**
> Określenie maxdelay > 0 może prowadzić do **ignorowania ważnych rekordów plików logów i pominiętych alertów**. Używaj go ostrożnie, na własne ryzyko, tylko wtedy, gdy jest to konieczne.

Domyślnie pozycje do monitorowania logów śledzą wszystkie nowe wiersze pojawiające się w plikach logów. Istnieją jednak aplikacje, które w niektórych sytuacjach zaczynają zapisywać ogromną liczbę komunikatów do swoich plików logów. Na przykład, jeśli baza danych lub serwer DNS jest niedostępny, takie aplikacje zalewają pliki logów tysiącami niemal identycznych komunikatów o błędach, dopóki normalne działanie nie zostanie przywrócone. Domyślnie wszystkie te komunikaty będą skrupulatnie analizowane, a pasujące wiersze będą wysyłane do serwera zgodnie z konfiguracją pozycji log i logrt.

Wbudowana ochrona przed przeciążeniem składa się z konfigurowalnego parametru maxlines (chroni serwer przed zbyt dużą liczbą przychodzących pasujących wierszy logów) oraz limitu 10*maxlines (chroni CPU hosta i operacje I/O przed przeciążeniem przez agent w ramach jednego sprawdzenia). Mimo to istnieją 2 problemy związane z wbudowaną ochroną. Po pierwsze, do serwera trafia duża liczba potencjalnie mało informacyjnych komunikatów, które zajmują miejsce w bazie danych. Po drugie, z powodu ograniczonej liczby wierszy analizowanych na sekundę agent może przez wiele godzin pozostawać w tyle za najnowszymi rekordami logów. Najprawdopodobniej wolisz zostać wcześniej poinformowany o bieżącej sytuacji w plikach logów, zamiast przez wiele godzin przeszukiwać stare rekordy.

Rozwiązaniem obu problemów jest użycie parametru maxdelay. Jeśli zostanie określone maxdelay > 0, podczas każdego sprawdzenia mierzona jest liczba przetworzonych bajtów, liczba pozostałych bajtów oraz czas przetwarzania. Na podstawie tych wartości agent oblicza szacowane opóźnienie - ile sekund zajęłoby przeanalizowanie wszystkich pozostałych rekordów w pliku logu.

Jeśli opóźnienie nie przekracza maxdelay, agent kontynuuje analizę pliku logu jak zwykle.

Jeśli opóźnienie jest większe niż maxdelay, agent **pomija fragment pliku logu, "przeskakując" nad nim** do nowej szacowanej pozycji, tak aby pozostałe wiersze mogły zostać przeanalizowane w ciągu maxdelay sekund.

Należy zauważyć, że agent nawet nie odczytuje pominiętych wierszy do bufora, lecz oblicza przybliżoną pozycję, do której ma przeskoczyć w pliku.

Fakt pominięcia wierszy pliku logu jest zapisywany w pliku logu agenta w następujący sposób:

```
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
```

Wartość „to byte" jest przybliżona, ponieważ po „przeskoku" agent dostosowuje pozycję w pliku do początku wiersza logu, który może znajdować się dalej w pliku lub wcześniej.

W zależności od tego, jak szybko rośnie plik w porównaniu z szybkością analizy pliku logu, możesz nie widzieć żadnych „przeskoków", rzadkie lub częste „przeskoki", duże lub małe „przeskoki", a nawet niewielki „przeskok" przy każdym sprawdzeniu. Wahania obciążenia systemu i opóźnienia sieciowe również wpływają na obliczanie opóźnienia, a tym samym na „przeskakiwanie" do przodu, aby nadążyć za parametrem maxdelay.

Nie zaleca się ustawiania maxdelay < update interval (może to skutkować częstymi małymi „przeskokami").

Uwagi dotyczące obsługi rotacji plików dziennika copytruncate

logrt z opcją copytruncate zakłada, że różne pliki dziennika mają różne rekordy (co najmniej ich znaczniki czasu są różne), dlatego sumy MD5 początkowych bloków (do pierwszych 512 bajtów) będą różne. Dwa pliki z takimi samymi sumami MD5 początkowych bloków oznaczają, że jeden z nich jest oryginałem, a drugi - kopią.

`logrt` z opcją `copytruncate` dokłada starań, aby poprawnie przetwarzać kopie plików dziennika bez zgłaszania duplikatów. Nie zaleca się jednak takich sytuacji, jak tworzenie wielu kopii pliku dziennika z tym samym znacznikiem czasu, rotacja pliku dziennika częściej niż wynosi interwał aktualizacji pozycja `logrt[]` oraz częste ponowne uruchamianie agent. Agent stara się rozsądnie obsługiwać wszystkie te sytuacje, ale nie można zagwarantować dobrych wyników w każdych okolicznościach.

Uwagi dotyczące trwałych plików dla pozycji log*[]

Cel plików trwałych

Gdy agent Zabbix jest uruchamiany, otrzymuje listę aktywnych kontroli z serwera Zabbix lub proxy. Dla metryk `log*[]` otrzymuje przetworzony rozmiar pliku dziennika oraz czas modyfikacji, aby ustalić, od którego miejsca rozpocząć monitorowanie pliku dziennika. W zależności od rzeczywistego rozmiaru pliku dziennika i czasu modyfikacji zgłoszonych przez system plików agent decyduje, czy kontynuować monitorowanie pliku dziennika od przetworzonego rozmiaru, czy ponownie przeanalizować plik dziennika od początku.

Działający agent utrzymuje większy zestaw atrybutów do śledzenia wszystkich monitorowanych plików dziennika między kontrolami. Ten stan przechowywany w pamięci jest tracony po zatrzymaniu agenta.

Nowy opcjonalny parametr `persistent_dir` określa katalog do zapisywania tego stanu pozycji `log[]`, `log.count[]`, `logrt[]` lub `logrt.count[]` w pliku. Stan pozycji dziennika jest przywracany z pliku trwałego po ponownym uruchomieniu agenta Zabbix.

Głównym przypadkiem użycia jest monitorowanie pliku dziennika znajdującego się na lustrzanym systemie plików. Do pewnego momentu plik dziennika jest zapisywany na obu lustrach. Następnie lustra są rozdzielane. Na aktywnej kopii plik dziennika nadal rośnie i otrzymuje nowe rekordy. Agent Zabbix analizuje go i wysyła przetworzony rozmiar pliku dziennika oraz czas modyfikacji do serwera. Na pasywnej kopii plik dziennika pozostaje bez zmian, daleko za aktywną kopią. Później system operacyjny i agent Zabbix są uruchamiane ponownie z pasywnej kopii. Przetworzony rozmiar pliku dziennika i czas modyfikacji, które agent Zabbix otrzymuje z serwera, mogą nie być prawidłowe dla sytuacji na pasywnej kopii. Aby kontynuować monitorowanie pliku dziennika od miejsca, w którym agent zakończył pracę w momencie rozdzielenia lustra systemu plików, agent przywraca swój stan z pliku trwałego.

Działanie agenta z plikiem trwałym

Podczas uruchamiania agent Zabbix nie wie nic o plikach trwałych. Dopiero po otrzymaniu listy aktywnych kontroli od serwer Zabbix (proxy) agent widzi, że niektóre pozycje dziennika powinny być obsługiwane przez pliki trwałe w określonych katalogach.

Podczas działania agenta pliki trwałe są otwierane do zapisu (za pomocą fopen(filename, ”w”)) i nadpisywane najnowszymi danymi. Prawdopodobieństwo utraty danych pliku trwałego, jeśli nadpisywanie i rozdzielenie kopii lustrzanej systemu plików nastąpią w tym samym czasie, jest bardzo małe; nie są wymagane żadne specjalne działania. Zapisywanie do pliku trwałego NIE jest poprzedzane wymuszonym synchronizowaniem z nośnikiem danych (`fsync()` nie jest wywoływane).

Nadpisanie najnowszymi danymi następuje po pomyślnym zgłoszeniu pasującego rekordu pliku dziennika lub metadanych (przetworzony rozmiar dziennika i czas modyfikacji) do serwer Zabbix. Może to następować nawet przy każdym sprawdzeniu pozycji, jeśli plik dziennika nadal się zmienia.

Nie są wykonywane żadne specjalne qnpónnqności podczas zamykania agenta.

Po otrzymaniu listy aktywnych kontroli agent oznacza przestarzałe pliki trwałe do usunięcia. Plik trwały staje się przestarzały, jeśli:

1. Odpowiednia pozycja logu nie jest już monitorowana.
2. Pozycja logu zostanie ponownie skonfigurowana z inną lokalizacją `persistent_dir` niż wcześniej.

Usunięcie jest wykonywane z opóźnieniem 24 godzin, ponieważ pliki logów w stanie NOTSUPPORTED nie są uwzględniane na liście aktywnych kontroli, ale mogą później stać się SUPPORTED, a ich pliki trwałe będą wtedy przydatne.

Jeśli agent zostanie zatrzymany przed upływem 24 godzin, przestarzałe pliki nie zostaną usunięte, ponieważ agent Zabbix nie otrzymuje już od serwera Zabbix informacji o ich lokalizacji.

Warning:
Ponowne skonfigurowanie `persistent_dir` pozycji logu z powrotem na starą lokalizację `persistent_dir` podczas zatrzymania agenta, bez usunięcia starego pliku trwałego przez użytkownika, spowoduje odtworzenie stanu agenta ze starego pliku trwałego, co doprowadzi do pominiętych komunikatów lub fałszywych alertów.

Nazewnictwo i lokalizacja plików trwałych

Zabbix agent rozróżnia aktywne sprawdzenia na podstawie ich kluczy. Na przykład `logrt[/home/zabbix/test.log]` i `logrt[/home/zabbix/test.log,]` to różne pozycje. Zmiana pozycji `logrt[/home/zabbix/test.log,,,10]` w frontend na `logrt[/home/zabbix/test.log,,,20]` spowoduje usunięcie pozycji `logrt[/home/zabbix/test.log,,,10]` z listy aktywnych sprawdzeń agenta oraz utworzenie pozycji `logrt[/home/zabbix/test.log,,,20]` (niektóre atrybuty są przenoszone podczas modyfikacji w frontend/serwerze, ale nie w agencie).

Nazwa pliku składa się z sumy MD5 klucza pozycji z dołączoną długością klucza pozycji, aby zmniejszyć prawdopodobieństwo kolizji. Na przykład stan pozycji `logrt[/home/zabbix50/test.log,,,,,,,,/home/zabbix50/agent_private]` będzie przechowywany w pliku trwałym c963ade4008054813bbc0a650bb8e09266.

Wiele pozycji log może używać tej samej wartości `persistent_dir`.

Wartość `persistent_dir` należy określić z uwzględnieniem konkretnego układu systemu plików, punktów montowania i opcji montowania oraz konfiguracji dublowania pamięci masowej - plik trwały powinien znajdować się w tym samym dublowanym systemie plików co monitorowany plik logu.

Jeśli katalog `persistent_dir` nie może zostać utworzony lub nie istnieje, albo jeśli Zabbix agent nie ma wystarczających uprawnień do tworzenia, zapisywania, odczytywania lub usuwania plików, pozycja logu otrzymuje status `NOTSUPPORTED`.

Jeśli podczas działania agent zostaną odebrane prawa dostępu do plików trwałego magazynu lub wystąpią inne błędy (np. brak miejsca na dysku), błędy zostaną zapisane w pliku dziennika agenta, ale pozycja dziennika nie otrzyma statusu `NOTSUPPORTED`.

Obciążenie I/O

Trwały plik pozycja jest aktualizowany po pomyślnym wysłaniu każdej partii danych (zawierającej dane pozycji) do serwer. Na przykład domyślna wartość `BufferSize` wynosi 100. Jeśli pozycja logu znalazła 70 pasujących rekordów, to pierwsze 50 rekordów zostanie wysłanych w jednej partii, trwały plik zostanie zaktualizowany, a następnie pozostałe 20 rekordów zostanie wysłanych (być może z pewnym opóźnieniem, gdy zostanie zgromadzonych więcej danych) w drugiej partii, po czym trwały plik zostanie ponownie zaktualizowany.

Działania w przypadku awarii komunikacji między agentem a serwerem

Każda pasująca linia z pozycji `log[]` i `logrt[]` oraz wynik każdego sprawdzenia pozycji `log.count[]` i `logrt.count[]` wymaga wolnego slotu w wyznaczonej 50% części bufora wysyłania agenta. Elementy bufora są regularnie wysyłane do serwer (lub proxy), a sloty bufora stają się ponownie wolne.

Dopóki w wyznaczonej części logów w buforze wysyłania agenta są wolne sloty i komunikacja między agentem a serwerem (lub proxy) jest przerwana, wyniki monitorowania logów są gromadzone w buforze wysyłania. Pomaga to ograniczyć skutki krótkotrwałych przerw w komunikacji.

Podczas dłuższych przerw w komunikacji wszystkie sloty logów zostają zajęte i podejmowane są następujące działania:

- sprawdzenia pozycji `log[]` i `logrt[]` są zatrzymywane. Gdy komunikacja zostanie przywrócona i w buforze będą dostępne wolne sloty, sprawdzenia są wznawiane od poprzedniego miejsca. Żadne pasujące linie nie są tracone, są jedynie raportowane później.
- sprawdzenia `log.count[]` i `logrt.count[]` są zatrzymywane, jeśli `maxdelay = 0` (domyślnie). Zachowanie jest podobne do pozycji `log[]` i `logrt[]` opisanych powyżej. Należy zauważyć, że może to wpływać na wyniki `log.count[]` i `logrt.count[]`: na przykład jedno sprawdzenie zlicza 100 pasujących linii w pliku logu, ale ponieważ w buforze nie ma wolnych slotów, sprawdzenie zostaje zatrzymane. Gdy komunikacja zostanie przywrócona, agent zlicza te same 100 pasujących linii, a także 70 nowych pasujących linii. Agent wysyła teraz count = 170, tak jakby zostały znalezione w jednym sprawdzeniu.
- sprawdzenia `log.count[]` i `logrt.count[]` z `maxdelay > 0`: jeśli podczas sprawdzenia nie nastąpił "skok", zachowanie jest podobne do opisanego powyżej. Jeśli nastąpił "skok" o linie pliku logu, zachowywana jest pozycja po "skoku", a wynik zliczania jest odrzucany. W ten sposób agent próbuje nadążać za rosnącym plikiem logu nawet w przypadku awarii komunikacji.

Obsługa błędów kompilacji i wykonania wyrażeń regularnych

Jeśli wyrażenie regularne używane w pozycjach `log[]`, `logrt[]`, `log.count[]` lub `logrt.count[]` nie może zostać skompilowane przez bibliotekę PCRE lub PCRE2, pozycja przechodzi w stan NIEOBSŁUGIWANA z komunikatem o błędzie. Aby kontynuować monitorowanie pozycji dziennika, należy poprawić wyrażenie regularne.

Jeśli wyrażenie regularne zostanie pomyślnie skompilowane, ale nie powiedzie się podczas wykonywania (dla niektórych lub wszystkich rekordów dziennika), pozycja dziennika pozostaje obsługiwana, a monitorowanie jest kontynuowane. Błąd wykonania jest zapisywany w pliku dziennika Zabbix agent (bez rekordu z pliku dziennika).

Częstotliwość zapisywania w dzienniku jest ograniczona do jednego błędu wykonania na sprawdzenie, aby umożliwić Zabbix agent monitorowanie własnego pliku dziennika. Na przykład jeśli analizowanych jest 10 rekordów, a dla 3 rekordów wystąpi błąd wykonania wyrażenia regularnego, w dzienniku agent zostanie zapisany jeden rekord.

Wyjątek: jeśli `MaxLinesPerSecond=1` i `update interval=1` (co pozwala analizować tylko jeden rekord podczas jednego sprawdzenia), błędy wykonania wyrażeń regularnych nie są zapisywane w dzienniku.

zabbix_agentd zapisuje klucz pozycji w przypadku błędu wykonania, natomiast zabbix_agent2 zapisuje identyfikator pozycji, aby ułatwić identyfikację pozycji dziennika, dla których występują błędy wykonania. W przypadku błędów wykonania zaleca się przeprojektowanie wyrażenia regularnego.

Prosty test

Przegląd

Proste testy są zwykle używane do zdalnego monitorowania usług bez użycia agenta.

Należy pamiętać, że agent Zabbix nie jest wymagany do prostych testów. Za przetwarzanie prostych testów odpowiada serwer/proxy Zabbix (nawiązywanie połączeń zewnętrznych itp.).

Przykłady użycia prostych testów:

```
net.tcp.service[ftp,,155]
net.tcp.service[http]
net.tcp.service.perf[http,,8080]
net.udp.service.perf[ntp]
```

Note:
Pola *Nazwa użytkownika* i *Hasło* (ograniczone do 255 znaków) w konfiguracji pozycji prostego testu są używane dla pozycji monitorowania VMware; w pozostałych przypadkach są ignorowane.

Obsługiwane sprawdzenia

Klucze pozycji są wymienione bez opcjonalnych parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Zobacz także klucze pozycji monitorowania VMware.

Klucz pozycji	Opis
icmpping	Dostępność hosta za pomocą ping ICMP.
icmppingloss	Procent utraconych pakietów.
icmppingretry	Dostępność hosta za pomocą ping ICMP z ponowieniami.
icmppingsec	Czas odpowiedzi ping ICMP.
net.tcp.service	Sprawdza, czy usługa działa i akceptuje połączenia TCP.
net.tcp.service.perf	Sprawdza wydajność usługi TCP.
net.udp.service	Sprawdza, czy usługa działa i odpowiada na żądania UDP.
net.udp.service.perf	Sprawdza wydajność usługi UDP.

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

icmpping[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

 Dostępność hosta za pomocą ping ICMP.
 Wartość zwracana: *0* - ping ICMP nie powiódł się; *1* - ping ICMP powiódł się.

Parametry:

- **target** - adres IP hosta lub nazwa DNS;
- **packets** - liczba pakietów;
- **interval** - czas między kolejnymi pakietami w milisekundach;
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **options** - używane do zezwalania na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Zobacz także tabelę wartości domyślnych.

Komentarz:

- Jeśli zostanie zwrócony co najmniej jeden z pakietów zdefiniowanych przez parametr `packets`, pozycja zwraca 1. Jeśli żaden z pakietów nie zostanie zwrócony, pozycja zwraca 0.

Przykład:

```
icmpping[,4] #Jeśli zostanie zwrócony co najmniej jeden z czterech pakietów, pozycja zwraca 1.
```

icmppingloss[<target>,<packets>,<interval>,<size>,<timeout>,<options>]

 Procent utraconych pakietów.
 Zwracana wartość: *Float*.

Parametry:

- **target** - adres IP hosta lub nazwa DNS;
- **packets** - liczba pakietów;
- **interval** - czas między kolejnymi pakietami w milisekundach;
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **options** - używane do zezwolenia na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Zobacz także tabelę wartości domyślnych.

icmppingretry[<target>,<retries>,<backoff>,<size>,<timeout>,<options>]

 Dostępność hosta za pomocą ICMP ping z ponowieniami. Jeśli pierwszy pakiet zakończy się powodzeniem, dalsze próby są wstrzymywane; jeśli pakiet nie powiedzie się, próby są ponawiane aż do osiągnięcia maksymalnej liczby prób określonej w parametrze `retries`. Ta pozycja jest przydatna do zmniejszenia liczby pakietów wysyłanych przez sieć.
 Wartość zwracana: *0* - ICMP ping nie powiódł się; *1* - ICMP ping zakończył się powodzeniem.

Parametry:

- **target** - adres IP lub nazwa DNS hosta;
- **retries** - liczba ponowień próby pingowania celu, bez uwzględnienia pierwszej próby (0 lub więcej; domyślnie 1);
- **backoff** - wartość, przez którą mnożony jest czas oczekiwania przy każdym kolejnym żądaniu (zakres 1.0-5.0; domyślnie 1.0);
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **options** - używane do zezwalania na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Zobacz także tabelę wartości domyślnych.

icmppingsec[<target>,<packets>,<interval>,<size>,<timeout>,<mode>,<options>]

 Czas odpowiedzi ping ICMP (w sekundach).
 Zwracana wartość: *Float*.

Parametry:

- **target** - adres IP lub nazwa DNS hosta;
- **packets** - liczba pakietów;
- **interval** - czas między kolejnymi pakietami w milisekundach;
- **size** - rozmiar pakietu w bajtach;
- **timeout** - limit czasu w milisekundach;
- **mode** - możliwe wartości: *min*, *max* lub *avg* (domyślnie);
- **options** - używane do zezwolenia na przekierowanie: jeśli puste (wartość domyślna), odpowiedzi przekierowane są traktowane jako niedostępność hosta docelowego; jeśli ustawione na *allow_redirect*, odpowiedzi przekierowane są traktowane jako dostępność hosta docelowego.

Uwagi:

- Pakiety utracone lub takie, dla których upłynął limit czasu, nie są uwzględniane w obliczeniach;
- Jeśli host jest niedostępny (osiągnięto limit czasu), pozycja zwróci 0;
- Jeśli zwracana wartość jest mniejsza niż 0.0001 sekundy, wartość zostanie ustawiona na 0.0001 sekundy;
- Zobacz także tabelę wartości domyślnych.

net.tcp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i akceptuje połączenia TCP.
 Zwracana wartość: *0* - usługa jest niedostępna; *1* - usługa działa.

Parametry:

- **service** - możliwe wartości: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Należy pamiętać, że dla usługi *tcp* podanie portu jest obowiązkowe;
- Te testy mogą powodować dodatkowe komunikaty w plikach dziennika demonów systemowych (zwykle rejestrowane są sesje SMTP i SSH);
- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście należy użyć `net.tcp.service[tcp,<ip>,port]` do takich testów.

Przykład:

```
net.tcp.service[ftp,,45] #Ta pozycja może być użyta do testowania dostępności serwera FTP na porcie TCP 45
```

Attention:
Jeśli SELinux działa w trybie enforced, niestandardowe proste testy TCP/UDP mogą być blokowane przez politykę. Aby zweryfikować i zezwolić na nowe połączenie wychodzące, sprawdź odmowy w audycie: używając `grep denied /var/log/audit/audit.log`

net.tcp.service.perf[service,<ip>,<port>]

 Sprawdza wydajność usługi TCP.
 Wartość zwracana: *Float*: *0.000000* - usługa jest niedostępna; *seconds* - liczba sekund poświęconych na połączenie z usługą.

Parametry:

- **service** - możliwe wartości: *ssh*, *ldap*, *smtp*, *ftp*, *http*, *pop*, *nntp*, *imap*, *tcp*, *https*, *telnet* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Uwagi:

- Należy pamiętać, że w przypadku usługi *tcp* podanie portu jest obowiązkowe;
- Sprawdzanie szyfrowanych protokołów (takich jak IMAP na porcie 993 lub POP na porcie 995) nie jest obecnie obsługiwane. Jako obejście należy użyć `net.tcp.service[tcp,<ip>,port]` do takich kontroli.

Przykład:

```
net.tcp.service.perf[ssh] #Ta pozycja może być użyta do testowania szybkości początkowej odpowiedzi serwer
```

net.udp.service[service,<ip>,<port>]

 Sprawdza, czy usługa działa i odpowiada na żądania UDP.
 Zwracana wartość: *0* - usługa jest niedostępna; *1* - usługa działa.

Parametry:

- **service** - możliwe wartości: *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
net.udp.service[ntp,,45] #Ta pozycja może być użyta do testowania dostępności usługi NTP na porcie UDP 45.
```

net.udp.service.perf[service,<ip>,<port>]

 Sprawdza wydajność usługi UDP.
 Zwracana wartość: *Float*: *0.000000* - usługa jest niedostępna; *seconds* - liczba sekund oczekiwania na odpowiedź z usługi.

Parametry:

- **service** - możliwe wartości: *ntp* (zobacz szczegóły);
- **ip** - adres IP lub nazwa DNS (domyślnie używany jest IP/DNS hosta);
- **port** - numer portu (domyślnie używany jest standardowy numer portu usługi).

Przykład:

```
net.udp.service.perf[ntp] #Ta pozycja może być użyta do testowania czasu odpowiedzi usługi NTP.
```

Attention:
Aby obsługa SourceIP była dostępna w prostych sprawdzeniach LDAP (np. `net.tcp.service[ldap]`), wymagana jest wersja OpenLDAP 2.6.1 lub nowsza.

Przetwarzanie limitu czasu

Elastyczne limity czasu pozycji, mimo że są obsługiwane dla prostych testów, nie mają zastosowania do pozycji `icmpping*` i VMware. Zobacz Obsługa elastycznych limitów czasu.

Pingi ICMP

Zabbix używa zewnętrznego narzędzia **fping** do przetwarzania pingów ICMP (**icmpping**, **icmppingloss**, **icmppingretry**, **icmppingsec**).

Instalacja

fping nie jest dołączony do Zabbix i musi zostać zainstalowany osobno:

- Różne platformy oparte na Uniksie mają pakiet fping w swoich domyślnych repozytoriach, ale nie jest on preinstalowany. W takim przypadku można użyć menedżera pakietów do zainstalowania fping.
- Zabbix udostępnia pakiety fping dla RHEL i jego pochodnych. Należy pamiętać, że pakiety te są dostarczane bez oficjalnego wsparcia.
- fping można również skompilować ze źródeł.

Konfiguracja

Określ lokalizację fping w parametrze *FpingLocation* pliku konfiguracyjnego serwer/proxy Zabbix (lub parametr *Fping6Location*, jeśli używasz adresów IPv6).

fping powinien być wykonywalny przez użytkownika, jako który działa serwer/proxy Zabbix, a użytkownik ten powinien mieć wystarczające uprawnienia.

Zobacz także: Znane problemy dotyczące przetwarzania prostych kontroli przy użyciu wersji fping poniżej 3.10.

Wartości domyślne

Wartości domyślne, limity i opis wartości parametrów sprawdzania ICMP:

Parametr	Jednostka	Opis	Flaga fping	Wartości domyślne ustawiane przez		Dozwolone limity przez Zabbix	
				fping	**Zabbix**	**min**	**max**
packets	liczba	Liczba pakietów żądań wysyłanych do celu	-C		3	1	10000
interval	milisekundy	Czas oczekiwania między kolejnymi pakietami wysyłanymi do pojedynczego celu	-p	1000		20	bez ograniczeń
size	bajty	Rozmiar pakietu w bajtach 56 bajtów na x86, 68 bajtów na x86_64	-b	56 lub 68		24	65507
timeout	milisekundy	**fping v3.x** - czas oczekiwania po wysłaniu ostatniego pakietu, zależny od flagi `-C` **fping v4.x** - indywidualny limit czasu dla każdego pakietu	-t	**fping v3.x** - 500 **fping v4.x** i nowsze - dziedziczony z flagi `-p`, ale nie większy niż 2000		50	bez ograniczeń
retries	liczba	Liczba ponowień próby wysłania polecenia ping do celu, z wyłączeniem pierwszej próby	-r	3	1	0	bez ograniczeń
backoff factor	liczba	Liczba, przez którą mnożony jest czas oczekiwania przy każdym kolejnym żądaniu	-B	1.5	1.0	1.0	5.0

Wartości domyślne mogą nieznacznie różnić się w zależności od platformy i wersji.

Ponadto Zabbix używa opcji fping `-i interval ms` (nie należy mylić jej z parametrem pozycji `interval` wymienionym w powyższej tabeli, który odpowiada opcji fping `-p`) oraz `-S source IP address` (lub `-I` w starszych wersjach fping). Opcje te są wykrywane automatycznie przez uruchamianie sprawdzeń z różnymi kombinacjami opcji. Zabbix próbuje wykryć minimalną wartość w milisekundach, którą fping dopuszcza dla opcji `-i`, testując trzy wartości: 0, 1 i 10. Pierwsza wartość, dla której sprawdzenie zakończy się powodzeniem, jest następnie używana podczas kolejnych sprawdzeń ICMP. Proces ten jest wykonywany niezależnie przez każdy proces ICMP pinger.

Automatycznie wykryte opcje fping tracą ważność co godzinę i są wykrywane ponownie przy następnej próbie wykonania sprawdzenia ICMP. Ustaw DebugLevel>=4, aby wyświetlać szczegóły tego procesu w pliku dziennika serwera lub proxy.

Zabbix zapisuje adresy IP, które mają zostać sprawdzone za pomocą dowolnych kluczy `icmpping*`, w pliku tymczasowym, który jest następnie przekazywany do fping. Jeśli pozycje mają różne parametry klucza, w jednym pliku zapisywane są tylko te, które mają identyczne parametry klucza. Wszystkie adresy IP zapisane w jednym pliku są sprawdzane przez fping równolegle, dlatego proces ICMP pinger Zabbix spędza stałą ilość czasu, niezależnie od liczby adresów IP w pliku.

Klucze pozycji monitorowania VMware

Lista kluczy pozycji monitorowania VMware została przeniesiona do sekcji Monitorowanie VMware.

Agent SNMP

Omówienie

Możesz chcieć używać monitorowania SNMP na urządzeniach takich jak drukarki, przełączniki sieciowe, routery lub UPS, które zwykle obsługują SNMP i na których próba skonfigurowania pełnych systemów operacyjnych oraz agentów Zabbix byłaby niepraktyczna.

Aby możliwe było pobieranie danych dostarczanych przez agentów SNMP na tych urządzeniach, serwer Zabbix musi zostać początkowo skonfigurowany z obsługą SNMP przez podanie flagi `--with-net-snmp`. Zaleca się również zainstalowanie plików MIB, aby zapewnić wyświetlanie wartości pozycji w prawidłowym formacie. Bez plików MIB mogą wystąpić problemy z formatowaniem, takie jak wyświetlanie wartości w HEX zamiast UTF-8 lub odwrotnie.

Kontrole SNMP są wykonywane wyłącznie przez protokół UDP.

Demon serwera Zabbix i proxy zapisują w logach wiersze podobne do poniższego, jeśli otrzymają nieprawidłową odpowiedź SNMP:

```
SNMP response from host "gateway" does not contain all of the requested variable bindings
```

Chociaż nie obejmują wszystkich problematycznych przypadków, są przydatne do identyfikowania pojedynczych urządzeń SNMP, dla których należy wyłączyć łączone żądania.

Serwer/proxy Zabbix ponowi próbę do 5 razy dla pozycji SNMP `walk` i `get`. Mechanizm ponawiania nie ma zastosowania do błędów rozwiązywania DNS.

W przypadku starszych kontroli SNMP (pojedynczy numer OID lub ciąg) serwer/proxy Zabbix ponowi próbę co najmniej jeden raz po nieudanej próbie zapytania: albo za pośrednictwem mechanizmu ponawiania biblioteki SNMP, albo poprzez wewnętrzny mechanizm łączonego przetwarzania.

Warning:
Jeśli monitorujesz urządzenia SNMPv3, upewnij się, że msgAuthoritativeEngineID (znany również jako snmpEngineID lub "Engine ID") nigdy nie jest współdzielony przez dwa urządzenia. Zgodnie z RFC 2571 (sekcja 3.1.1.1) musi on być unikalny dla każdego urządzenia.

Warning:
RFC3414 wymaga, aby urządzenia SNMPv3 zachowywały swoje engineBoots. Niektóre urządzenia tego nie robią, co powoduje, że ich wiadomości SNMP są odrzucane jako nieaktualne po ponownym uruchomieniu. W takiej sytuacji pamięć podręczna SNMP musi zostać ręcznie wyczyszczona na serwerze/proxy (za pomocą -R snmp_cache_reload) albo serwer/proxy musi zostać ponownie uruchomiony.

Note:
Zabbix buforuje mapowania SNMPv3 EngineID → IP i ponownie wykorzystuje zapisane w pamięci podręcznej EngineID dla kolejnych kontroli zamiast wysyłać za każdym razem sondę, co zmniejsza ruch sieciowy. Jeśli EngineID nie może zostać ponownie użyty, zostanie wykonana ponowna próba z sondą w celu wykrycia nowego EngineID.

Konfigurowanie monitorowania SNMP

Aby rozpocząć monitorowanie urządzenia za pomocą SNMP, należy wykonać następujące kroki:

Krok 1

Ustal ciąg SNMP (lub OID) pozycji, którą chcesz monitorować.

Aby uzyskać listę ciągów SNMP, użyj polecenia `snmpwalk` (część oprogramowania net-snmp, które powinno być zainstalowane jako element instalacji Zabbix) lub równoważnego narzędzia:

```
snmpwalk -v 2c -c public <host IP> .
```

Ponieważ 2c oznacza tutaj wersję SNMP, możesz również zastąpić ją wartością 1, aby wskazać na urządzeniu SNMP Version 1.

Powinno to zwrócić listę ciągów SNMP wraz z ich ostatnią wartością. Jeśli tak się nie stanie, możliwe, że społeczność SNMP ('community') różni się od standardowej wartości `public`; w takim przypadku trzeba będzie ją ustalić.

Następnie możesz przejrzeć listę, aż znajdziesz ciąg, który chcesz monitorować, np. jeśli chcesz monitorować bajty przychodzące do przełącznika na porcie 3, użyjesz ciągu `IF-MIB::ifHCInOctets.3` z tej linii:

```
IF-MIB::ifHCInOctets.3 = Counter64: 3409739121
```

Możesz teraz użyć polecenia `snmpget`, aby ustalić numeryczny OID dla `IF-MIB::ifHCInOctets.3`:

```
snmpget -v 2c -c public -On <host IP> IF-MIB::ifHCInOctets.3
```

Zwróć uwagę, że ostatnia liczba w ciągu to numer portu, który chcesz monitorować. Zobacz też: Dynamic indexes.

Powinno to zwrócić coś podobnego do poniższego:

```
.1.3.6.1.2.1.31.1.1.1.6.3 = Counter64: 3472126941
```

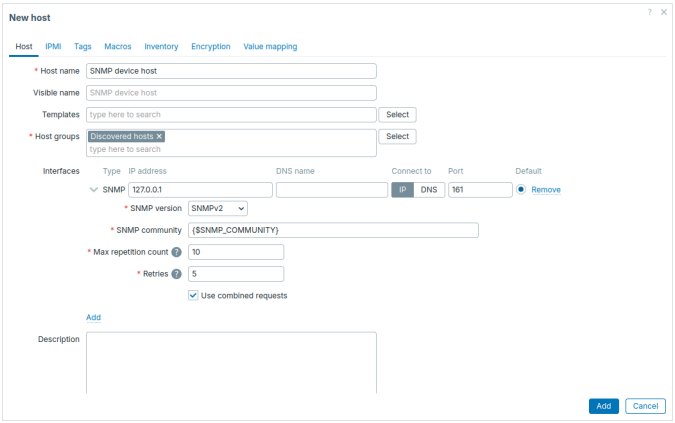
Ponownie, ostatnia liczba w OID to numer portu.

Note:
Niektóre z najczęściej używanych OID SNMP są automatycznie tłumaczone na reprezentację numeryczną przez Zabbix.

W ostatnim przykładzie typ wartości to "Counter64", który wewnętrznie odpowiada typowi ASN_COUNTER64. Pełna lista obsługiwanych typów to 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 oraz ASN_OBJECT_ID. Typy te w przybliżeniu odpowiadają "Counter32", "Counter64", "UInteger32", "INTEGER", "Float", "Double", "Timeticks", "Gauge32", "IpAddress", "OCTET STRING", "OBJECT IDENTIFIER" w wyniku **snmpget**, ale mogą być też wyświetlane jako "STRING", "Hex-STRING", "OID" i inne, w zależności od obecności wskazówki wyświetlania.

Krok 2

Utwórz host odpowiadający urządzeniu.



Dodaj interfejs SNMP dla hosta:

1. Wprowadź adres IP/nazwę DNS oraz numer portu.
2. Wybierz *wersję SNMP* z listy rozwijanej.
3. Dodaj dane uwierzytelniające interfejsu w zależności od wybranej wersji SNMP:
    - SNMPv1, v2 wymagają tylko community (zwykle 'public').
    - SNMPv3 wymaga bardziej szczegółowych opcji; zobacz SNMPv3.

4. Określ maksymalną wartość powtórzeń (domyślnie: 10) dla natywnych żądań zbiorczych SNMP (GetBulkRequest-PDUs); tylko dla pozycji `discovery[]` i `walk[]` w SNMPv2 i v3. Pamiętaj, że ustawienie tej wartości zbyt wysoko może spowodować przekroczenie limitu czasu sprawdzania agent SNMP.
5. Zaznacz pole wyboru *Use combined requests*, aby zezwolić na łączone przetwarzanie żądań SNMP (nie dotyczy natywnych zbiorczych żądań SNMP "walk" i "get").

Możesz użyć jednego z dostarczonych szablonów SNMP, który automatycznie doda zestaw pozycji. Przed użyciem szablonu sprawdź, czy jest on zgodny z hostem.

Kliknij *Add*, aby zapisać host.

SNMPv3

Następujące parametry są wymagane dla SNMPv3:

- *Context name* - Wprowadź nazwę kontekstu, aby zidentyfikować pozycja w podsieci SNMP.
Makra użytkownika są rozwijane w tym polu.
- *Security name* - Wprowadź nazwę zabezpieczeń.
Makra użytkownika są rozwijane w tym polu.
- *Security level* - Wybierz poziom zabezpieczeń:
  - *noAuthNoPriv* - nie są używane protokoły uwierzytelniania ani prywatności;
  - *AuthNoPriv* - używany jest protokół uwierzytelniania, protokół prywatności nie jest używany;
  - *AuthPriv* - używane są zarówno protokół uwierzytelniania, jak i protokół prywatności.
- *Authentication protocol* - Wybierz protokół uwierzytelniania: *MD5*, *SHA1*; w net-snmp 5.8 i nowszych także *SHA224*, *SHA256*, *SHA384* lub *SHA512*.
- *Authentication passphrase* - Wprowadź frazę hasła uwierzytelniania.
Makra użytkownika są rozwijane w tym polu.
- *Privacy protocol* - Wybierz protokół prywatności: *DES*, *AES128*, *AES192*, *AES256*, *AES192C* (Cisco) lub *AES256C* (Cisco).
Zobacz uwagi dotyczące obsługi protokołu prywatności.
- *Privacy passphrase* - Wprowadź frazę hasła prywatności.
Makra użytkownika są rozwijane w tym polu.

W przypadku nieprawidłowych poświadczeń SNMPv3 (security name, authentication protocol/passphrase, privacy protocol):

- Zabbix otrzymuje błąd ERROR z net-snmp, z wyjątkiem nieprawidłowej *Privacy passphrase*, w którym to przypadku Zabbix otrzymuje błąd TIMEOUT z net-snmp.
- Dostępność interfejsu SNMP zostanie przełączona na czerwony (niedostępny).

Note:
Zmiany w *Authentication protocol*, *Authentication passphrase*, *Privacy protocol* lub *Privacy passphrase*, wprowadzone bez zmiany *Security name*, są zwykle stosowane automatycznie po zaktualizowaniu odpowiedniego interfejsu SNMPv3 w Zabbix. W przypadkach, gdy zmieniany jest również *Security name*, wszystkie parametry zostaną zaktualizowane natychmiast.

Obsługa protokołów prywatności

W zależności od systemu operacyjnego i konfiguracji net-snmp, niektóre protokoły prywatności mogą być niedostępne:

- W niektórych nowszych systemach operacyjnych (na przykład RHEL9) obsługa DES została usunięta z pakietu `net-snmp`.
- Protokoły szyfrowania AES192 i silniejsze nie są obsługiwane od razu po instalacji na systemach starszych niż RHEL 8, CentOS 8, Oracle Linux 8, Debian 12, Ubuntu LTS 22.04, openSUSE Leap 15.5.

Aby sprawdzić, czy biblioteka net-snmp obsługuje AES192+, użyj jednej z poniższych opcji:

1. `net-snmp-config`:

```
net-snmp-config --configure-options
```

Jeśli wynik zawiera `--enable-blumenthal-aes`, AES192+ jest obsługiwane.

Należy pamiętać, że `net-snmp-config` jest częścią pakietu deweloperskiego dla SNMP (`libsnmp-dev` dla Debian/Ubuntu, `net-snmp-devel` dla CentOS/RHEL/OL/SUSE) i może nie być zainstalowany domyślnie.

2. `snmpget`:

```
snmpget -v 3 -x AES-256
```

Jeśli wynik zawiera `Invalid privacy protocol specified after -3x flag: AES-256`, AES192+ nie jest obsługiwane. Jeśli wynik zawiera `No hostname specified.`, AES192+ nie jest obsługiwane.

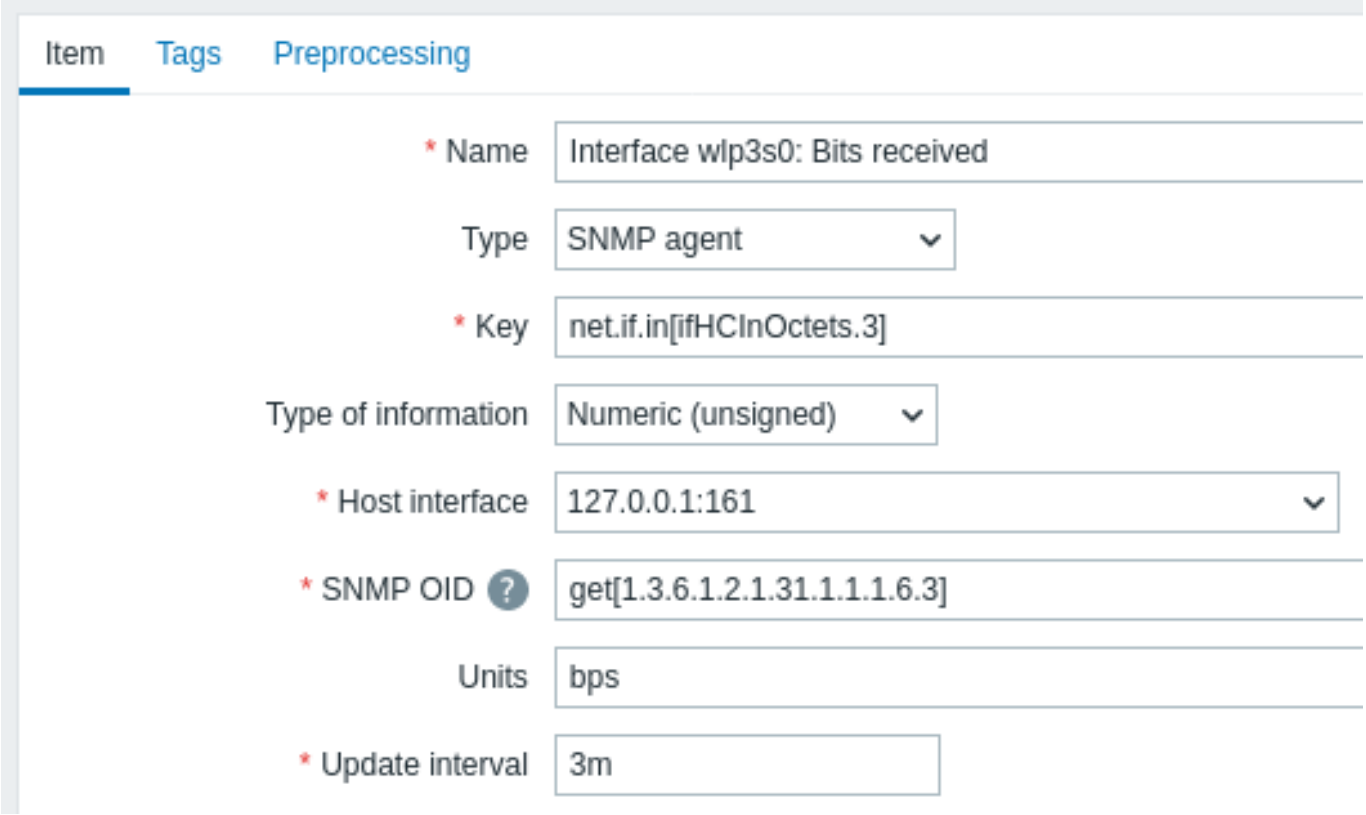
Jeśli Twoja biblioteka net-snmp nie obsługuje protokołów AES192 i wyższych, skompiluj ponownie net-snmp z opcją `--enable-blumenthal-a`
a następnie skompiluj ponownie serwer Zabbix, podając opcję `--with-net-snmp=/home/user/yourcustomnetsnmp/bin/net-snmp-co`

Krok 3

Utwórz pozycję do monitorowania.

Wróć teraz do Zabbix i kliknij *Pozycje* dla utworzonego wcześniej hosta SNMP. W zależności od tego, czy podczas tworzenia hosta użyto szablonu, będzie dostępna lista pozycji SNMP powiązanych z hostem albo tylko pusta lista. Załóżmy, że utworzysz pozycję samodzielnie, korzystając z informacji zebranych przed chwilą za pomocą snmpwalk i snmpget, dlatego kliknij *Utwórz pozycję*.

Wypełnij wymagane parametry w formularzu nowej pozycji:



Parametr	Opis
Nazwa	Wprowadź nazwę pozycji.
Typ	Wybierz tutaj **SNMP agent**.
Klucz	Wprowadź odpowiednio opisowy klucz.
Interfejs hosta	Upewnij się, że wybrano interfejs SNMP, np. przełącznika/routera.

Parametr	Opis
SNMP OID	Użyj jednego z obsługiwanych formatów, aby wprowadzić wartość lub wartości OID: **walk[OID1,OID2,...]** - pobiera poddrzewo wartości. Na przykład: `walk[1.3.6.1.2.1.2.2.1.2,1.3.6.1.2.1.2.2.1.3]`. Ta opcja korzysta asynchronicznie z natywnych żądań zbiorczych SNMP (GetBulkRequest-PDU). Ustawienia limitu czasu dla tej pozycji można skonfigurować w formularzu konfiguracji pozycji. Rozważ ustawienie niskiej wartości limitu czasu, aby uniknąć długich opóźnień, jeśli urządzenie jest niedostępne, ponieważ zostanie wykonanych do 5 ponownych prób, jeśli wcześniejsze próby przekroczą limit czasu lub zakończą się niepowodzeniem (np. limit czasu wynoszący 3 sekundy może skutkować oczekiwaniem trwającym 15 sekund). Możesz użyć tej pozycji jako pozycji nadrzędnej, z pozycjami zależnymi, które wyodrębniają dane z pozycji nadrzędnej za pomocą przetwarzania wstępnego. W jednym snmp walk można określić wiele OID, np. `walk[OID1,OID2,...]`, aby asynchronicznie przetwarzać po jednym OID naraz. Jeśli żądanie zbiorcze nie zwróci wyników, podejmowana jest próba pobrania pojedynczego rekordu bez żądania zbiorczego. Jako parametry można podawać nazwy MIB, dlatego `walk[1.3.6.1.2.1.2.2.1.2]` i `walk[ifDescr]` zwrócą ten sam wynik. Jeśli określono kilka OID/MIB, np. `walk[ifDescr,ifType,ifPhysAddress]`, wynik będzie połączoną listą. Żądania GetBulk są używane z interfejsami SNMPv2 i v3, a GetNext z interfejsami SNMPv1; maksymalna liczba powtórzeń dla żądań zbiorczych jest konfigurowana na poziomie interfejsu. Parametr maksymalnej liczby powtórzeń wpływa na żądania zbiorcze, określając maksymalną liczbę OID zwracanych w pojedynczej odpowiedzi zbiorczej. Wyższa wartość skutkuje większymi odpowiedziami zbiorczymi, zmniejszając liczbę wymaganych transmisji. Jednak nie wszystkie urządzenia mogą obsługiwać bardzo wysokie wartości, co może powodować problemy. Ta pozycja zwraca wynik narzędzia snmpwalk z parametrami -Oe -Ot -On. Możesz użyć tej pozycji jako pozycji nadrzędnej w wykrywaniu SNMP. **get[OID]** - asynchronicznie pobiera *pojedynczą* wartość. Na przykład: `get[1.3.6.1.2.1.31.1.1.1.6.3]` Ustawienia limitu czasu dla tej pozycji można skonfigurować w formularzu konfiguracji pozycji. Rozważ ustawienie niskiej wartości limitu czasu, aby uniknąć długich opóźnień, jeśli urządzenie jest niedostępne, ponieważ zostanie wykonanych do 5 ponownych prób, jeśli wcześniejsze próby przekroczą limit czasu lub zakończą się niepowodzeniem (np. limit czasu wynoszący 3 sekundy może skutkować oczekiwaniem trwającym 15 sekund). **OID** - (starsza opcja) wprowadź pojedynczy tekstowy lub numeryczny OID, aby synchronicznie pobrać pojedynczą wartość, opcjonalnie w połączeniu z innymi wartościami. Na przykład: `1.3.6.1.2.1.31.1.1.1.6.3`. W przypadku tej opcji limit czasu sprawdzania pozycji będzie równy wartości ustawionej w pliku konfiguracyjnym serwera. **Zaleca się** używanie pozycji `walk[OID]` i `get[OID]` w celu uzyskania lepszej wydajności. Wszystkie pozycje `walk[OID]` i `get[OID]` są wykonywane asynchronicznie - nie trzeba otrzymać odpowiedzi na jedno żądanie przed rozpoczęciem innych sprawdzeń. Rozpoznawanie DNS również odbywa się asynchronicznie. Maksymalna współbieżność sprawdzeń asynchronicznych wynosi 1000 (określana przez parametr MaxConcurrentChecksPerPoller). Liczbę asynchronicznych pollerów SNMP określa parametr StartSNMPPollers. Pamiętaj, że w przypadku statystyk ruchu sieciowego zwracanych przez dowolną z metod należy dodać krok *Zmiana na sekundę* na karcie *Przetwarzanie wstępne*; w przeciwnym razie z urządzenia SNMP zostanie pobrana wartość skumulowana zamiast ostatniej zmiany.

Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Zapisz teraz pozycję i przejdź do *Monitorowanie* > *Najnowsze dane*, aby wyświetlić dane SNMP.

Przykład 1

Ogólny przykład:

Parametr	Opis
OID	1.2.3.45.6.7.8.0 (lub .1.2.3.45.6.7.8.0)
Key	<Unikalny ciąg znaków używany jako odniesienie do wyzwalaczy> Na przykład „my_param”.

Zwróć uwagę, że OID może być podany zarówno w postaci numerycznej, jak i tekstowej. Jednak w niektórych przypadkach tekstowy OID musi zostać przekonwertowany do postaci numerycznej. W tym celu można użyć narzędzia snmpget:

```
snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

Przykład 2

Monitorowanie czasu działania:

Parametr	Opis
OID	MIB::sysUpTime.0
Key	router.uptime
Typ wartości	Float
Jednostki	uptime
Krok przetwarzania wstępnego: Mnożnik niestandardowy	0.01

Natywne żądania SNMP bulk

Pozycja **walk[OID1,OID2,...]** umożliwia korzystanie z natywnej funkcjonalności SNMP do obsługi żądań bulk (GetBulkRequest-PDU), dostępnej w wersjach SNMP 2/3.

Żądanie GetBulk w SNMP wykonuje wiele żądań GetNext i zwraca wynik w pojedynczej odpowiedzi. Można go używać zarówno dla zwykłych pozycji SNMP, jak i do wykrywania SNMP, aby zminimalizować liczbę operacji przesyłania danych w obie strony.

Pozycja SNMP **walk[OID1,OID2,...]** może być używana jako pozycja nadrzędna, która zbiera dane w jednym żądaniu, wraz z pozycjami zależnymi, które w razie potrzeby analizują odpowiedź za pomocą przetwarzania wstępnego.

Należy pamiętać, że korzystanie z natywnych żądań SNMP bulk nie jest związane z opcją łączenia żądań SNMP, która jest własnym sposobem Zabbix na łączenie wielu żądań SNMP (zobacz następną sekcję).

Dla pozycji SNMP bulk zostanie wykonanych maksymalnie pięć ponowień, aby uniknąć niepowodzenia w przypadku utraty jednego z pakietów. Limit czasu dla pozycji SNMP z `get` i `walk` (ustawiany w formularzu konfiguracji pozycji) obowiązuje dla całej sesji. Limit czasu jest stosowany niezależnie od tego, czy dane zostały pobrane w całości; jeśli dane zostaną odebrane częściowo (na przykład dane zostaną pomyślnie zebrane tylko dla jednego z wielu OID), pozycja stanie się nieobsługiwana z komunikatem „Only partial data received”. Po osiągnięciu limitu czasu nastąpi ponowienie, limit czasu zostanie zresetowany, a ostatnie żądanie zostanie wysłane ponownie, umożliwiając kontynuowanie sesji od ostatniego żądania, jeśli pojedynczy pakiet zostanie utracony lub dotrze zbyt późno. Rozważ ustawienie niskiej wartości limitu czasu, aby uniknąć długich opóźnień, jeśli urządzenie jest niedostępne, ponieważ w przypadku przekroczenia limitu czasu lub niepowodzenia wcześniejszych prób zostanie wykonanych maksymalnie 5 ponowień (np. limit czasu wynoszący 3 sekundy może skutkować oczekiwaniem trwającym 15 sekund).

Wewnętrzne działanie łącznego przetwarzania

serwer Zabbix i proxy mogą odpytywać urządzenia SNMP o wiele wartości w ramach jednego żądania. Dotyczy to kilku typów pozycji SNMP:

- zwykłe pozycje SNMP
- pozycje SNMP z dynamicznymi indeksami
- reguły wykrywania niskopoziomowego SNMP

Wszystkie pozycje SNMP na jednym interfejsie, z identycznymi parametrami, są planowane do odpytywania w tym samym czasie. Dwa pierwsze typy pozycji są pobierane przez pollery w partiach obejmujących maksymalnie 128 pozycji, natomiast reguły wykrywania niskopoziomowego są przetwarzane pojedynczo, tak jak wcześniej.

Na niższym poziomie podczas odpytywania wartości wykonywane są dwa rodzaje operacji: pobieranie wielu określonych obiektów oraz przechodzenie drzewa OID.

W przypadku „pobierania” używany jest GetRequest-PDU zawierający maksymalnie 128 powiązań zmiennych. W przypadku „przechodzenia” dla SNMPv1 używany jest GetNextRequest-PDU, a dla SNMPv2 i SNMPv3 — GetBulkRequest z polem „max-repetitions” o wartości maksymalnie 128.

Korzyści z łącznego przetwarzania dla poszczególnych typów pozycji SNMP są zatem następujące:

- zwykłe pozycje SNMP korzystają z usprawnień „pobierania”;
- pozycje SNMP z dynamicznymi indeksami korzystają zarówno z usprawnień „pobierania”, jak i „przechodzenia”: „pobieranie” jest używane do weryfikacji indeksu, a „przechodzenie” do budowania pamięci podręcznej;
- reguły wykrywania niskopoziomowego SNMP korzystają z usprawnień „przechodzenia”.

Istnieje jednak problem techniczny: nie wszystkie urządzenia mogą zwracać 128 wartości w jednym żądaniu. Niektóre zawsze zwracają prawidłową odpowiedź, ale inne zwracają błąd „tooBig(1)” albo w ogóle nie odpowiadają, gdy potencjalna odpowiedź przekracza określony limit.

Aby znaleźć optymalną liczbę obiektów, o które należy odpytywać dane urządzenie, Zabbix stosuje następującą strategię. Zaczyna ostrożnie od odpytywania jednej wartości w ramach żądania. Jeśli operacja się powiedzie, odpytywane są 2 wartości w ramach żądania. Jeśli ponownie się powiedzie, odpytywane są 3 wartości, a następnie proces jest kontynuowany w podobny sposób przez mnożenie liczby odpytywanych obiektów przez 1,5, co daje następującą sekwencję rozmiarów żądań: 1, 2, 3, 4, 6, 9, 13, 19, 28, 42, 63, 94, 128.

Jeśli jednak urządzenie odmówi udzielenia prawidłowej odpowiedzi, na przykład dla 42 zmiennych, Zabbix wykonuje dwie czynności.

Po pierwsze, dla bieżącej partii pozycji zmniejsza o połowę liczbę obiektów w jednym żądaniu i odpytywane jest 21 zmiennych. Jeśli urządzenie działa, zapytanie powinno zadziałać w zdecydowanej większości przypadków, ponieważ wiadomo, że 28 zmiennych działało, a 21 to znacznie mniej. Jeśli jednak to również się nie powiedzie, Zabbix przechodzi do odpytywania wartości pojedynczo. Jeśli i to się nie powiedzie, urządzenie zdecydowanie nie odpowiada, a rozmiar żądania nie jest przyczyną problemu.

Po drugie, dla kolejnych partii pozycji Zabbix zaczyna od ostatniej pomyślnie obsłużonej liczby zmiennych (w naszym przykładzie 28) i kontynuuje zwiększanie rozmiaru żądań o 1 aż do osiągnięcia limitu. Zakładając na przykład, że największy rozmiar odpowiedzi wynosi 32 zmienne, kolejne żądania będą miały rozmiary 29, 30, 31, 32 i 33. Ostatnie żądanie zakończy się niepowodzeniem, a Zabbix nigdy więcej nie wyśle żądania o rozmiarze 33. Od tego momentu Zabbix będzie odpytywać to urządzenie o maksymalnie 32 zmienne.

Jeśli duże zapytania zakończą się niepowodzeniem przy tej liczbie zmiennych, może to oznaczać jedną z dwóch rzeczy. Nie można znać dokładnych kryteriów używanych przez urządzenie do ograniczania rozmiaru odpowiedzi, ale próbujemy je przybliżyć za pomocą liczby zmiennych. Pierwsza możliwość jest taka, że liczba zmiennych jest zbliżona do rzeczywistego limitu rozmiaru odpowiedzi urządzenia w typowych przypadkach: czasami odpowiedź jest mniejsza od limitu, a czasami go przekracza. Druga możliwość jest taka, że pakiet UDP w jednym z kierunków po prostu został utracony. Z tych powodów, jeśli Zabbix otrzyma nieudaną odpowiedź na zapytanie, zmniejsza maksymalną liczbę zmiennych, aby wejść głębiej w bezpieczny zakres urządzenia, ale robi to maksymalnie dwa razy.

W powyższym przykładzie, jeśli zapytanie z 32 zmiennymi zakończy się niepowodzeniem, Zabbix zmniejszy ich liczbę do 31. Jeśli to zapytanie również się nie powiedzie, Zabbix zmniejszy liczbę do 30. Zabbix nie zmniejszy jednak liczby poniżej 30, ponieważ założy, że kolejne niepowodzenia wynikają z utraty pakietów UDP, a nie z ograniczenia urządzenia.

Jeśli jednak urządzenie nie może prawidłowo obsługiwać łącznych żądań z innych powodów, a opisana powyżej heurystyka nie działa, dla każdego interfejsu dostępne jest ustawienie „Use combined requests”, które pozwala wyłączyć łączne żądania dla tego urządzenia.

Jeśli łączne żądania powodują częściowe lub nieprawidłowo sformułowane odpowiedzi, prowadzące do niepoprawnych obliczeń na sekundę (delta), na przykład pozornych skoków liczników interfejsu, wyłącz opcję *Use combined requests* dla danego interfejsu, aby wymusić oddzielne zapytania dla każdej pozycji; często zapobiega to fałszywym skokom. Alternatywnie rozważ użycie asynchronicznych pozycji `get[]` lub `walk[]`, które są wykonywane asynchronicznie i nie podlegają grupowaniu *Use combined requests* na poziomie interfejsu — można ich używać zamiast starszych synchronicznych kontroli OID, aby uniknąć problemów związanych z łącznymi żądaniami. Aby zidentyfikować urządzenia, których dotyczy problem, poszukaj w dziennikach serwera/proxy wpisów podobnych do przedstawionego w sekcji Przegląd.

Ponadto, jeśli interfejs często staje się niedostępny, może być konieczne zwiększenie parametru `UnavailableDelay` w plikach konfiguracyjnych serwera Zabbix lub proxy Zabbix, aby zmniejszyć częstotliwość żądań. Pozycje mogą stać się nieobsługiwane, jeśli podczas wykrywania lub przechodzenia OID zostaną odebrane niepełne dane.

Dynamiczne indeksy

Przegląd

Chociaż wymagany numer indeksu (na przykład interfejsu sieciowego) można znaleźć wśród OID-ów SNMP, czasami nie można całkowicie polegać na tym, że numer indeksu zawsze pozostanie taki sam.

Numery indeksów mogą być dynamiczne — mogą zmieniać się z upływem czasu, a w konsekwencji Twoja pozycja może przestać działać.

Aby uniknąć takiej sytuacji, można zdefiniować OID, który uwzględnia możliwość zmiany numeru indeksu.

Na przykład, jeśli chcesz pobrać wartość indeksu do dołączenia do **ifInOctets**, która odpowiada interfejsowi **GigabitEthernet0/1** na urządzeniu Cisco, użyj następującego OID:

```
ifInOctets["index","ifDescr","GigabitEthernet0/1"]
```

Składnia

Używana jest specjalna składnia dla OID:

<OID danych>[”index”,”<bazowy OID indeksu>”,”<szukany ciąg znaków>”]

Parametr	Opis
OID danych	Główny OID używany do pobierania danych dla pozycji.
index	Metoda przetwarzania. Obecnie obsługiwana jest jedna metoda: **index** - wyszukuje indeks i dołącza go do OID danych
bazowy OID indeksu	Ten OID zostanie sprawdzony w celu uzyskania wartości indeksu odpowiadającej ciągowi znaków.
szukany ciąg znaków	Ciąg znaków używany do dokładnego dopasowania wartości podczas wyszukiwania. Wielkość liter ma znaczenie.

Przykład

Pobieranie użycia pamięci procesu *apache*.

Jeśli używana jest ta składnia OID:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem["index","HOST-RESOURCES-MIB::hrSWRunPath", "/usr/sbin/apache2"]
```

numer indeksu zostanie tutaj wyszukany:

```
...
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"
...
```

Teraz mamy indeks 5388. Indeks zostanie dołączony do OID danych, aby otrzymać interesującą nas wartość:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem.5388 = INTEGER: 31468 KBytes
```

Buforowanie wyszukiwania indeksów

Gdy żądana jest pozycja dynamicznego indeksu, Zabbix pobiera i buforuje całą tabelę SNMP pod bazowym OID dla indeksu, nawet jeśli dopasowanie zostałoby znalezione wcześniej. Jest to wykonywane na wypadek, gdyby inna pozycja odwoływała się później do tego samego bazowego OID — Zabbix wyszuka indeks w pamięci podręcznej zamiast ponownie odpytywać monitorowany host. Należy pamiętać, że każdy proces pollera używa oddzielnej pamięci podręcznej.

We wszystkich kolejnych operacjach pobierania wartości weryfikowany jest tylko znaleziony indeks. Jeśli nie uległ zmianie, żądana jest wartość. Jeśli uległ zmianie, pamięć podręczna jest przebudowywana — każdy poller, który napotka zmieniony indeks, ponownie przechodzi tabelę indeksów SNMP.

Specjalne OID-y

Niektóre z najczęściej używanych OID-ów SNMP są automatycznie tłumaczone przez Zabbix na reprezentację numeryczną. Na przykład **ifIndex** jest tłumaczone na **1.3.6.1.2.1.2.2.1.1**, a **ifIndex.0** jest tłumaczone na **1.3.6.1.2.1.2.2.1.1.0**.

Tabela zawiera listę specjalnych OID-ów.

Special OID	Identifier	Description
ifIndex	1.3.6.1.2.1.2.2.1.1	Unikalna wartość dla każdego interfejsu.
ifDescr	1.3.6.1.2.1.2.2.1.2	Tekstowy ciąg zawierający informacje o interfejsie. Ciąg ten powinien zawierać nazwę producenta, nazwę produktu oraz wersję sprzętowego interfejsu.
ifType	1.3.6.1.2.1.2.2.1.3	Typ interfejsu, określony zgodnie z fizycznym/protokółem łącza bezpośrednio ’poniżej’ warstwy sieciowej w stosie protokołów.
ifMtu	1.3.6.1.2.1.2.2.1.4	Rozmiar największego datagramu, który może zostać wysłany / odebrany na interfejsie, określony w oktetach.

Special OID	Identifier	Description
ifSpeed	1.3.6.1.2.1.2.2.1.5	Szacunkowa bieżąca przepustowość interfejsu w bitach na sekundę.
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	Adres interfejsu na warstwie protokołu bezpośrednio ‘poniżej’ warstwy sieciowej w stosie protokołów.
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	Bieżący stan administracyjny interfejsu.
ifOperStatus	1.3.6.1.2.1.2.2.1.8	Bieżący stan operacyjny interfejsu.
ifInOctets	1.3.6.1.2.1.2.2.1.10	Łączna liczba oktetów odebranych na interfejsie, w tym znaki ramek.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	Liczba pakietów unicast sieci podrzędnej dostarczonych do protokołu wyższej warstwy.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	Liczba pakietów nie-unicastowych (tj. broadcast lub multicast sieci podrzędnej) dostarczonych do protokołu wyższej warstwy.
ifInDiscards	1.3.6.1.2.1.2.2.1.13	Liczba pakietów przychodzących, które zostały odrzucone, mimo że nie wykryto błędów uniemożliwiających ich dostarczenie do protokołu wyższej warstwy. Jednym z możliwych powodów odrzucenia takiego pakietu może być zwolnienie miejsca w buforze.
ifInErrors	1.3.6.1.2.1.2.2.1.14	Liczba pakietów przychodzących zawierających błędy, które uniemożliwiły ich dostarczenie do protokołu wyższej warstwy.
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15	Liczba pakietów odebranych przez interfejs, które zostały odrzucone z powodu nieznanego lub nieobsługiwanego protokołu.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	Łączna liczba oktetów wysłanych z interfejsu, w tym znaki ramek.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	Łączna liczba pakietów, których wysłanie zostało zażądane przez protokoły wyższych warstw i które nie były adresowane do adresu multicast ani broadcast na tej podwarstwie, w tym tych, które zostały odrzucone lub nie zostały wysłane.
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	Łączna liczba pakietów, których wysłanie zostało zażądane przez protokoły wyższych warstw i które były adresowane do adresu multicast lub broadcast na tej podwarstwie, w tym tych, które zostały odrzucone lub nie zostały wysłane.
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	Liczba pakietów wychodzących, które zostały odrzucone, mimo że nie wykryto błędów uniemożliwiających ich wysłanie. Jednym z możliwych powodów odrzucenia takiego pakietu może być zwolnienie miejsca w buforze.
ifOutErrors	1.3.6.1.2.1.2.2.1.20	Liczba pakietów wychodzących, których nie można było wysłać z powodu błędów.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	Długość kolejki pakietów wyjściowych (w pakietach).

Pliki MIB

Wprowadzenie

MIB oznacza bazę informacji zarządzania (Management Information Base). Pliki MIB umożliwiają używanie tekstowych reprezentacji OID (identyfikatorów obiektów). Podczas monitorowania urządzeń SNMP za pomocą Zabbix można używać surowych OID, ale jeśli wygodniejsze jest używanie reprezentacji tekstowej, należy zainstalować pliki MIB.

Na przykład,

```
ifHCOutOctets
```

jest tekstową reprezentacją OID

```
1.3.6.1.2.1.31.1.1.1.10
```

Instalacja plików MIB

W systemach opartych na Debianie:

```
apt install snmp-mibs-downloader
download-mibs
```

W systemach opartych na RedHat:

```
dnf install net-snmp-libs
```

Włączanie plików MIB

W systemach opartych na RedHat pliki MIB powinny być domyślnie włączone. W systemach opartych na Debian należy edytować plik `/etc/snmp/snmp.conf` i zakomentować linię `mibs :`

```
# Ponieważ pakiety snmp są dostarczane bez plików MIB z powodów licencyjnych, ładowanie
# plików MIB jest domyślnie wyłączone. Jeśli dodano pliki MIB, można ponownie włączyć
# ich ładowanie, zakomentowując poniższą linię.
mibs :
```

Testowanie plików MIB

Testowanie SNMP MIB można wykonać za pomocą narzędzia `snmpwalk`. Jeśli nie jest ono zainstalowane, skorzystaj z poniższych instrukcji.

W systemach opartych na Debianie:

```
apt install snmp
```

W systemach opartych na RedHat:

```
dnf install net-snmp-utils
```

Następnie poniższe polecenie nie powinno zwracać błędu podczas odpytywania urządzenia sieciowego:

```
$ snmpwalk -v 2c -c public <NETWORK DEVICE IP> 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
[...]
```

Używanie MIB-ów w Zabbixie

Najważniejsze, o czym należy pamiętać, to fakt, że procesy Zabbix nie są informowane o zmianach wprowadzonych w plikach MIB. Dlatego po każdej zmianie należy ponownie uruchomić serwer lub proxy Zabbix, na przykład:

```
systemctl restart zabbix-server
```

Po tym zmiany wprowadzone w plikach MIB zaczną obowiązywać.

Używanie niestandardowych plików MIB

Istnieją standardowe pliki MIB dostarczane z każdą dystrybucją GNU/Linux. Jednak niektórzy producenci urządzeń udostępniają własne.

Załóżmy, że chcesz użyć pliku MIB CISCO-SMI. Poniższe instrukcje pobiorą go i zainstalują:

```
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
```

Teraz powinno być możliwe jego użycie. Spróbuj przetłumaczyć nazwę obiektu *ciscoProducts* z pliku MIB na OID:

```
snmptranslate -IR -On CISCO-SMI::ciscoProducts
.1.3.6.1.4.1.9.1
```

Jeśli zamiast OID otrzymasz błędy, upewnij się, że wszystkie poprzednie polecenia zostały wykonane bez błędów.

Tłumaczenie nazwy obiektu zadziałało, więc możesz już używać niestandardowego pliku MIB. Zwróć uwagę na prefiks nazwy MIB (*CISCO-SMI::*) użyty w zapytaniu. Będzie on potrzebny zarówno podczas korzystania z narzędzi wiersza poleceń, jak i Zabbix.

Nie zapomnij zrestartować serwera/proxy Zabbix przed użyciem tego pliku MIB w Zabbix.

> **Attention:**
> Pamiętaj, że pliki MIB mogą mieć zależności. Oznacza to, że jeden MIB może wymagać innego. Aby spełnić te zależności, musisz zainstalować wszystkie powiązane pliki MIB.

SNMP trap

Przegląd

Odbieranie trapów SNMP jest przeciwieństwem odpytywania urządzeń obsługujących SNMP.

W tym przypadku informacja jest wysyłana z urządzenia obsługującego SNMP do `snmptrapd` i jest zbierana lub „przechwytywana" przez serwer Zabbix lub proxy Zabbix z pliku.

Zazwyczaj trapy są wysyłane po zmianie pewnego warunku, a agent łączy się z serwerem na porcie 162 (w przeciwieństwie do portu 161 po stronie agenta, który jest używany do zapytań). Używanie trapów może wykrywać krótkotrwałe problemy, które występują pomiędzy interwałami odpytywania i mogą zostać pominięte przez dane z zapytań.

Odbieranie trapów SNMP w Zabbix zostało zaprojektowane do współpracy z `snmptrapd` oraz jednym z mechanizmów przekazywania trapów do Zabbix - skryptem Bash lub Perl albo SNMPTT.

Note:
Najprostszym sposobem skonfigurowania monitorowania trapów po skonfigurowaniu Zabbix jest użycie rozwiązania ze skryptem Bash, ponieważ Perl i SNMPTT często są nieobecne w nowoczesnych dystrybucjach i wymagają bardziej złożonej konfiguracji. Jednak to rozwiązanie używa skryptu skonfigurowanego jako `traphandle`. Dla lepszej wydajności w systemach produkcyjnych użyj wbudowanego rozwiązania Perl (albo skryptu z opcją do `perl`, albo SNMPTT).

Przepływ pracy odbierania trapa:

1. `snmptrapd` odbiera trap.
2. `snmptrapd` przekazuje trap do skryptu odbierającego (Bash, Perl) lub SNMPTT.
3. Odbiornik analizuje, formatuje i zapisuje trap do pliku.
4. Zabbix SNMP trapper odczytuje i analizuje plik trapów.
5. Dla każdego trapa Zabbix znajduje wszystkie pozycje SNMP trapper z interfejsami hosta pasującymi do odebranego adresu trapa. Należy zauważyć, że podczas dopasowywania używany jest tylko wybrany *IP* lub *DNS* w interfejsie hosta.
6. Dla każdej znalezionej pozycji trap jest porównywany z wyrażeniem regularnym w `snmptrap[regexp]`. Trap jest ustawiany jako wartość **wszystkich** dopasowanych pozycji. Jeśli nie zostanie znaleziona żadna pasująca pozycja i istnieje pozycja `snmptrap.fallback`, trap jest ustawiany jako jej wartość.
7. Jeśli trap nie został ustawiony jako wartość żadnej pozycji, Zabbix domyślnie rejestruje niedopasowany trap. (Jest to konfigurowane przez opcję *Log unmatched SNMP traps* w *Administration* > *General* > *Other*.)

Uwagi dotyczące przełączenia awaryjnego HA

Podczas przełączania węzła wysokiej dostępności (HA) Zabbix będzie kontynuował przetwarzanie od ostatniego rekordu w ramach ostatniego znacznika czasu ISO 8601; jeśli ten sam rekord nie zostanie znaleziony, do określenia ostatniej pozycji zostanie użyty tylko znacznik czasu.

Konfigurowanie trapów SNMP

Ten typ pozycji wymaga następującej konfiguracji frontend:

Krok 1: Utwórz interfejs SNMP dla hosta

1. W *Data collection* > *Hosts*, utwórz host lub edytuj istniejący.
2. W polu *Interfaces* wybierz typ interfejsu *SNMP*.
3. Wprowadź adres IP/nazwę DNS oraz numer portu.
4. Wybierz *wersję SNMP* z listy rozwijanej: (*SNMPv1*, *SNMPv2* lub *SNMPv3*) i dodaj dane uwierzytelniające interfejsu odpowiednio do wybranej wersji SNMP.

Adres z każdego odebranego trapu zostanie porównany z adresami IP i DNS wszystkich interfejsów SNMP, aby znaleźć odpowiadające hosty.

Krok 2: Skonfiguruj pozycję

1. Dla hosta utwórz pozycję lub edytuj istniejącą.
2. W polu *Key* użyj jednego z kluczy trapów SNMP:
   - `snmptrap[regexp]` - przechwytuje wszystkie trapy SNMP pasujące do wyrażenia regularnego określonego w parametrze `regexp`; jeśli `regexp` nie jest określony, przechwytuje dowolny trap.
 W parametrze są obsługiwane makra użytkownika i globalne wyrażenia regularne.
 Wartość zwracana: trap SNMP.
 Tę pozycję można ustawić tylko dla interfejsów SNMP.
   - `snmptrap.fallback` - przechwytuje wszystkie trapy SNMP, które nie zostały przechwycone przez żadną z pozycji `snmptrap[]` dla tego interfejsu.
 Wartość zwracana: trap SNMP.
 Tę pozycję można ustawić tylko dla interfejsów SNMP.

Note:
Obecnie nie jest obsługiwane dopasowywanie wielowierszowych wyrażeń regularnych.

3. Ustaw *Type of information* na *Log*, aby znaczniki czasu mogły zostać przeanalizowane. Inne formaty, takie jak *Numeric*, są również akceptowalne, ale mogą wymagać niestandardowego modułu obsługi trapów.

Konfigurowanie monitorowania pułapek SNMP

Konfigurowanie serwera/proxy Zabbix

Aby odczytywać pułapki, serwer Zabbix lub proxy muszą być skonfigurowane tak, aby uruchamiały proces SNMP trapper i wskazywały plik pułapek, do którego zapisuje SNMPTT lub odbiornik pułapek Bash/Perl. W tym celu należy edytować plik konfiguracyjny (`zabbix_server.conf` lub `zabbix_proxy.conf`):

```
StartSNMPTrapper=1
SNMPTrapperFile=[TRAP FILE]
```

Warning:
Jeśli używany jest parametr systemd `PrivateTmp`, ten plik prawdopodobnie nie będzie działał w `/tmp`.

Konfigurowanie odbiornika pułapek Bash

Wymagania: tylko snmptrapd.

Skrypt odbiornika pułapek Bash script może być używany do przekazywania pułapek do serwera Zabbix z snmptrapd za pomocą pliku trappera. Aby go skonfigurować, dodaj opcję `traphandle` do pliku konfiguracyjnego snmptrapd (`snmptrapd.conf`), zobacz przykład.

Note:
Może być konieczne ponowne uruchomienie snmptrapd, aby uwzględnił zmiany w swojej konfiguracji.

Konfigurowanie odbiornika trapów Perl

Wymagania: Perl, Net-SNMP skompilowany z `--enable-embedded-perl` (domyślnie od Net-SNMP 5.4)

Odbiornik trapów Perl (szukaj `misc/snmptrap/zabbix_trap_receiver.pl`) może być używany do przekazywania trapów do serwer Zabbix bezpośrednio z snmptrapd. Aby go skonfigurować:

- Dodaj skrypt Perl do pliku konfiguracyjnego snmptrapd (`snmptrapd.conf`), np.:

```
perl do "[FULL PATH TO PERL RECEIVER SCRIPT]";
```

- Skonfiguruj odbiornik, np.:

```
$SNMPTrapperFile = '[TRAP FILE]';
$DateTimeFormat = '[DATE TIME FORMAT]';
```

Note:
snmptrapd może wymagać ponownego uruchomienia, aby uwzględnić zmiany w swojej konfiguracji.

Note:
Jeśli nazwa skryptu nie jest ujęta w cudzysłów, snmptrapd odmówi uruchomienia i wyświetli komunikaty podobne do poniższych:


```
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
```

Konfigurowanie SNMPTT

Na początku należy skonfigurować snmptrapd do używania SNMPTT.

Note:
Dla najlepszej wydajności SNMPTT powinien być skonfigurowany jako demon z użyciem `snmptthandler-embedded`, aby przekazywać do niego pułapki. Instrukcje dotyczące konfigurowania SNMPTT.

Gdy SNMPTT jest skonfigurowany do odbierania pułapek, skonfiguruj `snmptt.ini`:

1. Włącz użycie modułu Perl z pakietu `NET-SNMP`:

```
net_snmp_perl_enable = 1
```

2. Zapisuj pułapki do pliku pułapek, który będzie odczytywany przez Zabbix:

```
log_enable = 1
log_file = [TRAP FILE]
```

3. Ustaw format daty i czasu:

```
date_time_format = %Y-%m-%dT%H:%M:%S%z
```

Warning:
Pakiet `net-snmp-perl` został usunięty w RHEL 8.0-8.2; ponownie dodany w RHEL 8.3. Więcej informacji znajduje się w sekcji znane problemy.

Teraz sformatuj pułapki tak, aby Zabbix mógł je rozpoznać (edytuj `snmptt.conf`):

1. Każda instrukcja `FORMAT` powinna zaczynać się od `ZBXTRAP [address]`, gdzie `[address]` będzie porównywany z adresami IP i DNS interfejsów SNMP w Zabbix. Np.:

```
EVENT coldStart .1.3.6.1.6.3.1.1.5.1 "Status Events" Normal
FORMAT ZBXTRAP $aA Device reinitialized (coldStart)
```

2. Więcej informacji o formacie pułapek SNMP znajduje się poniżej.

Attention:
Nie używaj nieznanych pułapek - Zabbix nie będzie w stanie ich rozpoznać. Nieznane pułapki można obsłużyć, definiując ogólne zdarzenie w `snmptt.conf`:


```
EVENT general .* "General event" Normal
```

Format trapu SNMP

Wszystkie niestandardowe odbiorniki trapów Perl oraz konfiguracja trapów SNMPTT muszą formatować trap w następujący sposób:

```
[timestamp] [the trap, part 1] ZBXTRAP [address] [the trap, part 2]
```

gdzie

- `[timestamp]` - znacznik czasu w formacie `%Y-%m-%dT%H:%M:%S%z`.
- `ZBXTRAP` - nagłówek wskazujący, że w tym wierszu zaczyna się nowy trap.
- `[address]` - adres IP używany do znalezienia hosta dla tego trapu.

Zwróć uwagę, że `ZBXTRAP` i `[address]` zostaną usunięte z komunikatu podczas przetwarzania. Jeśli trap ma inny format, Zabbix może zinterpretować trapy w nieoczekiwany sposób.

Przykładowy 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
```

Spowoduje to utworzenie następującego trapu dla interfejsu SNMP z 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
```

Wymagania systemowe

Note:
Zaleca się zainstalowanie plików MIB, aby zapewnić, że wartości pozycji będą wyświetlane w prawidłowym formacie. Bez plików MIB mogą wystąpić problemy z formatowaniem, takie jak wyświetlanie wartości w formacie HEX zamiast UTF-8 lub odwrotnie.

Obsługa dużych plików

Zabbix obsługuje duże pliki dla plików SNMP trapper. Maksymalny rozmiar pliku, który Zabbix może odczytać, wynosi 2^63 (8 EiB). Należy pamiętać, że system plików może narzucać niższy limit rozmiaru pliku.

Rotacja logów

Zabbix nie zapewnia własnego mechanizmu rotacji logów — powinien być on obsługiwany przez użytkownika. Rotacja logów powinna najpierw zmienić nazwę starego pliku, a dopiero później go usunąć, aby nie utracić żadnych trapów:

1. Zabbix otwiera plik trapów w ostatnio znanej lokalizacji i przechodzi do kroku 3.
2. Zabbix sprawdza, czy aktualnie otwarty plik został zrotowany, porównując numer inode z numerem inode zdefiniowanego pliku trapów. Jeśli nie ma otwartego pliku, Zabbix resetuje ostatnią lokalizację i przechodzi do kroku 1.

3. Zabbix odczytuje dane z aktualnie otwartego pliku i ustawia nową lokalizację.
4. Nowe dane są analizowane. Jeśli był to zrotowany plik, plik zostaje zamknięty i następuje powrót do kroku 2.
5. Jeśli nie było nowych danych, Zabbix usypia na 1 sekundę i wraca do kroku 2.

System plików

Ze względu na implementację pliku trap, Zabbix wymaga, aby system plików obsługiwał i-węzły, aby rozróżniać pliki (informacje są pobierane za pomocą wywołania `stat()`).

Przykłady konfiguracji z użyciem różnych wersji protokołu SNMP

W tym przykładzie użyto snmptrapd oraz skryptu odbiorczego Bash do przekazywania trapów do serwera Zabbix.

Konfiguracja:

1. Skonfiguruj Zabbix tak, aby uruchamiał SNMP trapper i używał określonego pliku trapów. Dodaj do `zabbix_server.conf`:

```
StartSNMPTrapper=1
SNMPTrapperFile=/var/lib/zabbix/snmptraps/snmptraps.log
```

2. Pobierz skrypt Bash do `/usr/sbin/zabbix_trap_handler.sh`:

```
curl -o /usr/sbin/zabbix_trap_handler.sh https://raw.githubusercontent.com/zabbix/zabbix-docker/trunk/temp
```

W razie potrzeby dostosuj zmienną `ZABBIX_TRAPS_FILE` w skrypcie. Aby użyć wartości domyślnej, najpierw utwórz katalog nadrzędny:

```
mkdir -p /var/lib/zabbix/snmptraps
```

3. Dodaj poniższy wpis do `snmtrapd.conf` (zobacz działający przykład):

```
traphandle default /bin/bash /usr/sbin/zabbix_trap_handler.sh
```

> **Note:**
> Może być konieczne ponowne uruchomienie snmptrapd, aby wczytał zmiany w konfiguracji.

4. Utwórz pozycję SNMP TEST (pamiętaj o początkowych wymaganiach dotyczących konfiguracji):

   Typ: *SNMP trap*
 Typ informacji: *Log*
 Interfejs hosta: *SNMP 127.0.0.1*
 Klucz: `snmptrap["linkUp"]`
 Format czasu w logu: `yyyyMMdd.hhmmss`

Pamiętaj, że używany jest format daty i czasu ISO 8601.

5. Następnie skonfigurujemy `snmptrapd` dla wybranej wersji protokołu SNMP i wyślemy testowe trapy za pomocą narzędzia `snmptrap`.

SNMPv1, SNMPv2

Protokoły SNMPv1 i SNMPv2 korzystają z uwierzytelniania za pomocą „community string". W poniższym przykładzie użyjemy wartości „secret" jako community string. Tę samą wartość należy ustawić na nadawcach trapów SNMP.

Należy pamiętać, że mimo powszechnego stosowania w środowiskach produkcyjnych SNMPv2 nie zapewnia szyfrowania ani rzeczywistego uwierzytelniania nadawcy. Dane są przesyłane jawnym tekstem, dlatego tych wersji protokołu należy używać wyłącznie w bezpiecznych środowiskach, takich jak sieci prywatne, i nigdy w publicznych ani należących do innych podmiotów sieciach.

Wersja SNMP 1 nie jest obecnie praktycznie używana, ponieważ nie obsługuje liczników 64-bitowych i jest uznawana za protokół starszego typu.

Aby włączyć odbieranie trapów SNMPv1 lub SNMPv2, dodaj poniższy wiersz do `snmptrapd.conf`. Zastąp `secret` community string SNMP skonfigurowanym na nadawcach trapów SNMP:

```
authCommunity log,execute,net secret
```

Następnie możemy wysłać testowy trap za pomocą `snmptrap`. W tym przykładzie użyjemy popularnego identyfikatora OID „link up":

```
snmptrap -v 2c -c secret localhost '' linkUp.0
```

SNMPv3

SNMPv3 rozwiązuje problemy bezpieczeństwa SNMPv1/v2 oraz zapewnia uwierzytelnianie i szyfrowanie. Można używać metod uwierzytelniania MD5 lub różnych metod SHA, a także DES lub różnych wariantów AES jako szyfru.

Skonfiguruj `snmptrapd` tak, aby akceptował trapy SNMPv3, dodając poniższe wiersze do `snmptrapd.conf`:

```
##### Utwórz użytkownika SNMPv3 USM z uwierzytelnianiem i ochroną prywatności
createUser -e <engineID> <securityName> SHA <authKey> AES <privKey>
##### Zezwól użytkownikowi na rejestrowanie trapów i wykonywanie skryptów obsługi trapów
authUser log,execute <securityName>
```

Na przykład:

```
createUser -e 0x8000000001020304 traptest SHA myAuthPassword AES myPrivPassword
authUser log,execute traptest
```

Attention:
Zwróć uwagę na słowo kluczowe „execute", które umożliwia wykonywanie skryptów dla tego modelu zabezpieczeń użytkownika.

Aby przetestować konfigurację trapów SNMPv3, wyślij testowy trap za pomocą następującego polecenia:

```
snmptrap -v 3 -n "" -a SHA -A myAuthPassword -x AES -X myPrivPassword -l authPriv -u traptest -e 0x800000
```

Warning:
Jeśli chcesz używać silnych metod szyfrowania, takich jak AES192 lub AES256, użyj net-snmp w wersji 5.8 lub nowszej. Może być konieczna ponowna kompilacja z opcją `configure: --enable-blumenthal-aes`. Starsze wersje net-snmp nie obsługują AES192/AES256. Zobacz także: Strong Authentication or Encryption.

Weryfikacja

W obu przykładach w pliku `/var/lib/zabbix/snmptraps/snmptraps.log` zobaczysz podobne wiersze:

```
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
```

Wartość pozycji w Zabbix będzie następująca:

```
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
```

Przykład z użyciem Perla:

```
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
```

Zobacz także

- Artykuł na blogu Zabbix o pułapkach SNMP
- Konfigurowanie snmptrapd (oficjalna dokumentacja net-snmp)
- Konfigurowanie snmptrapd do odbierania powiadomień SNMPv3 (oficjalna dokumentacja net-snmp)

Wewnętrzne Zabbix

Przegląd

Kontrole wewnętrzne umożliwiają monitorowanie wewnętrznych procesów Zabbix. Innymi słowy, można monitorować działanie serwera Zabbix lub proxy Zabbix.

Kontrole wewnętrzne są obliczane:

- na serwerze Zabbix - jeśli host jest monitorowany przez serwer
- na proxy Zabbix - jeśli host jest monitorowany przez proxy

Kontrole wewnętrzne są przetwarzane przez serwer lub proxy niezależnie od stanu konserwacji hosta.

Aby użyć tej pozycji, wybierz typ pozycji **Zabbix internal**.

Note:
Kontrole wewnętrzne są przetwarzane przez pollery Zabbix.

Wydajność

Używanie niektórych wewnętrznych pozycji może negatywnie wpływać na wydajność. Są to następujące pozycje:

- `zabbix[host,,items]`
- `zabbix[host,,items_unsupported]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[queue,,]`
- `zabbix[requiredperformance]`
- `zabbix[stats,,,queue,,]`
- `zabbix[triggers]`

Dotyczy to również sekcji frontend Informacje o systemie oraz Kolejka.

Obsługiwane kontrole

Klucze pozycji są wymienione bez konfigurowalnych parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby wyświetlić pełne informacje.

Klucz pozycji	Opis
zabbix[apm,]	Statystyki danych telemetrycznych otrzymanych przez kolektor APM.
zabbix[boottime]	Czas uruchomienia procesu serwera Zabbix lub proxy Zabbix, w sekundach.
zabbix[cluster,discovery,nodes]	Wykrywa węzły klastra wysokiej dostępności.
zabbix[connector_queue]	Liczba wartości umieszczonych w kolejce connectora.
zabbix[discovery_queue]	Liczba kontroli sieciowych umieszczonych w kolejce wykrywania.
zabbix[host,,items]	Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych) na hoście.
zabbix[host,,items_unsupported]	Liczba włączonych nieobsługiwanych pozycji na hoście.
zabbix[host,,maintenance]	Bieżący stan konserwacji hosta.
zabbix[host,active_agent,available]	Dostępność kontroli wykonywanych przez aktywnego agent na hoście.
zabbix[host,discovery,interfaces]	Szczegóły wszystkich skonfigurowanych interfejsów hosta w frontend Zabbix.
zabbix[host,,available]	Dostępność głównego interfejsu dla określonego typu kontroli na hoście.
zabbix[hosts]	Liczba monitorowanych hostów.
zabbix[items]	Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych).
zabbix[items_unsupported]	Liczba nieobsługiwanych pozycji.
zabbix[java,]	Informacje o Java gateway Zabbix.
zabbix[lld_queue]	Liczba wartości umieszczonych w kolejce przetwarzania wykrywania niskopoziomowego.
zabbix[preprocessing]	Statystyki wartości otrzymanych przez menedżera wstępnego przetwarzania.
zabbix[preprocessing_queue]	Liczba wartości umieszczonych w kolejce wstępnego przetwarzania.
zabbix[process,,]	Procent czasu, jaki określony proces Zabbix lub grupa procesów (identyfikowana przez <type> i <mode>) spędziła w stanie <state>.
zabbix[proxy,]	Informacje o proxy Zabbix.
zabbix[proxy,discovery]	Lista proxy Zabbix.
zabbix[proxy group,available]	Liczba dostępnych proxy w grupie proxy.
zabbix[proxy group,pavailable]	Odsetek dostępnych proxy w grupie proxy.
zabbix[proxy group,proxies]	Lista proxy Zabbix w grupie proxy.
zabbix[proxy group,state]	Stan grupy proxy.

Klucz pozycji	Opis
zabbix[proxy group,discovery]	Zwraca listę grup proxy wraz z danymi konfiguracyjnymi i danymi w czasie rzeczywistym.
zabbix[proxy_buffer,buffer,]	Zwraca statystyki wykorzystania bufora pamięci proxy.
zabbix[proxy_buffer,state,changes]	Zwraca liczbę zmian stanu między trybami bufora dyskowego i pamięciowego od momentu uruchomienia.
zabbix[proxy_buffer,state,current]	Zwraca bieżący stan pracy, w którym przechowywane są nowe dane.
zabbix[proxy_history]	Liczba wartości w tabeli historii proxy oczekujących na wysłanie do serwera.
zabbix[queue,]	Liczba monitorowanych pozycji w kolejce, które są opóźnione o co najmniej <from> sekund, ale o mniej niż <to> sekund.
zabbix[rcache,]	Statystyki dostępności pamięci podręcznej konfiguracji Zabbix.
zabbix[requiredperformance]	Wymagana wydajność serwera Zabbix lub proxy Zabbix, wyrażona jako oczekiwana liczba nowych wartości na sekundę.
zabbix[stats,]	Zwraca wewnętrzne metryki serwera Zabbix lub proxy. Jeśli określono <ip> i <port>, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
zabbix[stats,,queue,]	Zwraca wewnętrzne metryki kolejki serwera Zabbix lub proxy. Jeśli określono <ip> i <port>, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z instancji lokalnej.
zabbix[tcache,]	Statystyki skuteczności pamięci podręcznej funkcji trendów Zabbix.
zabbix[triggers]	Liczba włączonych wyzwalaczy w bazie danych Zabbix, ze wszystkimi pozycjami włączonymi na włączonych hostach.
zabbix[uptime]	Czas działania procesu serwera Zabbix lub proxy, w sekundach.
zabbix[vcache,buffer,]	Statystyki dostępności pamięci podręcznej wartości Zabbix.
zabbix[vcache,cache,]	Statystyki skuteczności pamięci podręcznej wartości Zabbix.
zabbix[version]	Wersja serwera Zabbix lub proxy.
zabbix[vmware,buffer,]	Statystyki dostępności pamięci podręcznej vmware Zabbix.
zabbix[vps,written]	Łączna liczba wartości historii zapisanych w bazie danych.
zabbix[wcache,]	Statystyki i dostępność pamięci podręcznej zapisu Zabbix.

Szczegóły klucza pozycji

- Parametry bez nawiasów ostrych są obowiązkowe i muszą być użyte *dokładnie w tej postaci* (na przykład „host” i „available” w `zabbix[host,<type>,available]`).
- Parametry ujęte w nawiasy ostre **< >** muszą zostać zastąpione prawidłową wartością. Jeśli parametr ma wartość domyślną, można go pominąć.
- Wartości dla pozycji i parametrów pozycji oznaczonych jako „not supported on proxy” mogą być pobierane tylko wtedy, gdy host jest monitorowany przez serwer. Z kolei wartości oznaczone jako „not supported on server” mogą być pobierane tylko wtedy, gdy host jest monitorowany przez proxy.

zabbix[apm,<type>]

 Statystyki danych telemetrycznych odebranych przez kolektor APM:

- *requests* - liczba żądań odebranych przez kolektor APM (licznik):
  - *accepted* - liczba żądań zaakceptowanych do przetwarzania;
  - *dropped* - liczba odrzuconych żądań;
- *rows* - liczba wierszy danych telemetrycznych zapisanych w źródle danych, według typu danych (licznik).

Zwracana wartość: *obiekt JSON*.

Parametry:

- **type** - *traces*, *metrics* lub *logs*. Jeśli nie określono, zwracane są statystyki dla wszystkich typów.

Uwagi:

- Ta pozycja nie jest obsługiwana na serwerze Zabbix.

Przykład zwracanych wartości dla `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]

 Czas uruchomienia procesu serwera Zabbix lub procesu Zabbix proxy w sekundach.
 Zwracana wartość: *Integer*.

zabbix[cep]

 Complex event processing statistics.
 Return value: *JSON*.

zabbix[cluster,discovery,nodes]

 Wykrywa węzły klastra wysokiej dostępności.
 Zwracana wartość: *obiekt JSON*.

Uwagi:

- Tej pozycji można używać w wykrywaniu niskopoziomowym.
- Zwracane pola w większości odpowiadają właściwościom obiektu węzła wysokiej dostępności.
- Dodatkowe pola:
  - `db_timestamp` - bieżący czas serwera (znacznik czasu Unix) zawarty w odpowiedzi
  - `lastaccess_age` - liczba sekund od ostatniego sygnału heartbeat węzła (db_timestamp - lastaccess)
  - `status` - stan węzła:
    - 0 - w trybie gotowości
    - 1 - zatrzymany ręcznie
    - 2 - niedostępny
    - 3 - aktywny

Przykład zwracanych wartości:

```
[
  {
    "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]

 Liczba wartości umieszczonych w kolejce connector queue.
 Zwracana wartość: *Integer*.

zabbix[discovery_queue]

 Liczba kontroli sieciowych umieszczonych w kolejce wykrywania.
 Wartość zwracana: *Integer*.

zabbix[host,„items]

 Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych) na hoście.
 Zwracana wartość: *Integer*.

zabbix[host,„items_unsupported]

 Liczba włączonych nieobsługiwanych pozycji na hoście.
 Zwracana wartość: *Integer*.

zabbix[host,„maintenance]

 Bieżący status konserwacji hosta.
 Zwracane wartości: *0* - stan normalny; *1* - konserwacja z gromadzeniem danych; *2* - konserwacja bez gromadzenia danych.

Uwagi:

- Ta pozycja jest zawsze przetwarzana przez serwer Zabbix niezależnie od lokalizacji hosta (na serwerze lub proxy). Proxy nie otrzyma tej pozycji wraz z danymi konfiguracyjnymi.
- Drugi parametr musi być pusty i jest zarezerwowany do wykorzystania w przyszłości.

zabbix[host,active_agent,available]

 Dostępność aktywnych kontroli agenta na hoście.
 Zwracane wartości: *0* - nieznana; *1* - dostępny; *2* - niedostępny.

zabbix[host,discovery,interfaces]

 Szczegóły wszystkich skonfigurowanych interfejsów hosta w Zabbix frontend.
 Zwracana wartość: *obiekt JSON*.

Uwagi:

- Ta pozycja może być używana w odkrywaniu niskiego poziomu.
- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbix[host,<type>,available]

 Dostępność głównego interfejsu określonego typu kontroli na hoście.
 Zwracane wartości: *0* - niedostępny; *1* - dostępny; *2* - nieznany.

Parametry:

- **type** - *agent*, *snmp*, *ipmi* lub *jmx*.

Komentarze:

- Wartość pozycji jest obliczana zgodnie z parametrami konfiguracyjnymi dotyczącymi nieosiągalności/niedostępności hosta.

zabbixhosts

 Liczba monitorowanych hostów.
 Zwracana wartość: *Integer*.

zabbix[items]

 Liczba włączonych pozycji (obsługiwanych i nieobsługiwanych).
 Zwracana wartość: *Integer*.

zabbix[items_unsupported]

 Liczba nieobsługiwanych pozycji.
 Zwracana wartość: *Integer*.

zabbix[java,„<param>]

 Informacje o Zabbix Java gateway.
 Zwracane wartości: *1* - jeśli `<param>` to *ping*; *wersja Java gateway* - jeśli `<param>` to *version* (na przykład: ”8.0.0”).

Parametry:

- **param** - *ping* lub *version*.

Komentarze:

- Ta pozycja może być używana do sprawdzania dostępności Java gateway za pomocą funkcji wyzwalacza `nodata()`.
- Drugi parametr musi być pusty i jest zarezerwowany do przyszłego użycia.

zabbix[lld_queue]

 Liczba wartości umieszczonych w kolejce przetwarzania wykrywania niskiego poziomu.
 Zwracana wartość: *Integer*.

Uwagi:

- Ta pozycja może być używana do monitorowania długości kolejki przetwarzania wykrywania niskiego poziomu.

zabbix[preprocessing]

 Statystyki wartości otrzymanych przez menedżera preprocessing:

- *queued* — liczba i rozmiar wartości oczekujących w kolejce, które wymagają preprocessing (licznik)
- *direct* — liczba i rozmiar wartości oczekujących w kolejce, które nie wymagają preprocessing (licznik)
- *queue* — liczba wartości umieszczonych w kolejce preprocessing (to samo co `zabbix[preprocessing_queue]`)

Zwracana wartość: *JSON*.

Przykład zwracanych wartości:

```
{"data":
    {
    "queued": {
        "count": 106,
        "size": 58620
    },
    "direct": {
        "count": 395,
        "size": 33843
    },
    "queue": 0
    }
}
```

zabbix[preprocessing_queue]

 Liczba wartości umieszczonych w kolejce przetwarzania wstępnego.
 Zwracana wartość: *Integer*.

Uwagi:

- Ta pozycja może być używana do monitorowania długości kolejki przetwarzania wstępnego.

zabbix[process,<type>,<mode>,<state>]

 Procent czasu, który określony proces Zabbix lub grupa procesów (identyfikowana przez `<type>` i `<mode>`) spędziła w stanie `<state>`. Wartość jest obliczana wyłącznie dla ostatniej minuty.
 Zwracana wartość: *Float*.

Parametry:

- **type**:
  - dla procesów serwera: *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*;
  - dla procesów 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* - średnia wartość dla wszystkich procesów danego typu (domyślnie);
  - *count* - zwraca liczbę procesów danego typu, `<state>` nie należy określać;
  - *max* - wartość maksymalna;
  - *min* - wartość minimalna;
  - *<process number>* - numer procesu (od 1 do liczby instancji utworzonych wstępnie; na przykład jeśli działają 4 procesy trapper, wartość mieści się w zakresie od 1 do 4);
- **state**:
  - *busy* - proces jest zajęty, na przykład przetwarza żądanie (domyślnie);
  - *idle* - proces jest bezczynny i nic nie robi.

Uwagi:

- Jeśli <mode> jest numerem procesu Zabbix, który nie działa (na przykład gdy działa 5 procesów poller, a jako <mode> określono 6), taka pozycja stanie się nieobsługiwana.
- Wartość minimalna i maksymalna odnoszą się do procentowego wykorzystania pojedynczego procesu. Jeśli więc w grupie 3 procesów poller wartości procentowe dla poszczególnych procesów wynosiły 2, 18 i 66, wartość minimalna wyniesie 2, a maksymalna 66.
- Procesy raportują w pamięci współdzielonej, co robią, a proces samomonitorowania podsumowuje te dane co sekundę. Zmiany stanu (busy/idle) są rejestrowane w momencie zmiany - dlatego proces, który staje się zajęty, rejestruje ten stan i nie zmienia go ani nie aktualizuje, dopóki nie stanie się bezczynny. Dzięki temu nawet całkowicie zawieszone procesy będą prawidłowo rejestrowane jako zajęte w 100%.
- Obecnie „busy" oznacza „nie śpi", ale w przyszłości mogą zostać wprowadzone dodatkowe stany - oczekiwanie na blokady, wykonywanie zapytań do bazy danych itp. Należy pamiętać, że asynchroniczne procesy poller są uznawane za zajęte, jeśli osiągnęły limit określony przez parametr konfiguracyjny MaxConcurrentChecksPerPoller serwera/proxy.
- W systemie Linux i większości innych systemów rozdzielczość wynosi 1/100 sekundy.

Przykłady:

```
zabbix[process,poller,avg,busy] #średni czas, który procesy poller spędziły na wykonywaniu operacji w ciąg
zabbix[process,"icmp pinger",max,busy] #maksymalny czas, który dowolny proces ICMP pinger spędził na wykon
zabbix[process,"history syncer",2,busy] #czas, który drugi proces history syncer spędził na wykonywaniu op
zabbix[process,trapper,count] #liczba aktualnie działających procesów trapper
```

zabbix[proxy,<name>,<param>]

 Informacje o proxy Zabbix.
 Zwracana wartość: *Integer*.

Parametry:

- **name** - nazwa proxy;
- **param**:
  - *lastaccess* - znacznik czasu ostatniego komunikatu heartbeat otrzymanego z proxy;
  - *delay* - czas, przez jaki zebrane wartości pozostają niewysłane; obliczany jako „opóźnienie proxy" + („bieżący czas serwera" - „lastaccess proxy"), gdzie „opóźnienie proxy" to różnica między bieżącym czasem proxy a znacznikiem czasu najstarszej niewysłanej wartości na proxy.

Uwagi:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.
- Funkcji `fuzzytime()` można użyć do sprawdzenia dostępności proxy.

Przykład:

```
zabbix[proxy,"Germany",lastaccess] #znacznik czasu ostatniego komunikatu heartbeat otrzymanego z proxy „Ge
```

zabbix[proxy,discovery]

 Lista proxy Zabbix zawierająca nazwę, tryb, szyfrowanie, kompresję, wersję, czas ostatniego wykrycia, liczbę hostów, liczbę pozycji, wymaganą liczbę wartości na sekundę (vps), stan wersji (aktualna/nieaktualna/nieobsługiwana), limity czasu według typu pozycji, nazwę grupy proxy (jeśli proxy należy do grupy), stan (nieznany/nieaktywne/aktywne).
 Zwracana wartość: *obiekt JSON*.

Komentarz:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.

Przykład zwracanych wartości:

```
[
  {
    "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]

 Liczba dostępnych proxy w grupie proxy.
 Zwracana wartość: *Integer*.

Parametry:

- **name** - nazwa grupy proxy.

Komentarz:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.

zabbix[proxy group,<name>,pavailable]

 Procent dostępnych online proxy w grupie proxy.
 Zwracana wartość: *Float*.

Parametry:

- **name** - nazwa grupy proxy.

Komentarz:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.

zabbix[proxy group,<name>,proxies]

 Lista proxy Zabbix w grupie proxy zawierająca nazwę, tryb, szyfrowanie, kompresję, wersję, czas ostatniego kontaktu, liczbę hostów, liczbę pozycji, wymaganą liczbę wartości na sekundę (vps), status wersji (aktualna/nieaktualna/nieobsługiwana), limity czasu, nazwę grupy proxy oraz stan (nieznany/wyłączony online).
 Zwracana wartość: *JSON*.

Parametry:

- **name** - nazwa grupy proxy.

Komentarz:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.

zabbix[proxy group,<name>,state]

 Stan grupy proxy.
 Zwracana wartość: *0* - nieznany; *1* - offline; *2* - odzyskiwanie; *3* - online; *4* - pogarszanie.

Parametry:

- **name** - nazwa grupy proxy.

Komentarz:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.

zabbix[proxy group,discovery]

 Zwraca listę grup proxy wraz z danymi konfiguracyjnymi i danymi w czasie rzeczywistym. Dane konfiguracyjne obejmują nazwę grupy proxy, opóźnienie przełączania awaryjnego oraz minimalną wymaganą liczbę dostępnych proxy. Dane w czasie rzeczywistym obejmują stan grupy proxy (szczegółowe informacje znajdują się w komentarzach), liczbę dostępnych proxy oraz odsetek dostępnych proxy.
 Wartość zwracana: *JSON*.

Komentarze:

- Ta pozycja nie jest obsługiwana na hostach monitorowanych przez proxy lub grupę proxy.
- Ta pozycja nie zwraca proxy nieprzypisanych do grupy.
- Jeśli wartość `failover_delay` lub `min_online` jest nieprawidłowa, zgłaszana jest specjalna wartość −1, aby to wskazać. Nieprawidłowe wartości mogą wystąpić, jeśli do konfiguracji użyto makr, a makra nie mogą zostać rozwinięte do prawidłowej wartości.
- Stan grupy proxy jest zgłaszany jako liczba całkowita: *0* - nieznany; *1* - offline; *2* - odzyskiwanie; *3* - online; *4* - pogarszanie stanu.

Przykład zwracanych wartości:

```
{
    "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>]

 Statystyki wykorzystania bufora pamięci proxy.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla wartości procentowej).

Parametry:

- **mode**:
  - *total* - całkowity rozmiar bufora (można użyć do sprawdzenia, czy bufor pamięci jest włączony);
  - *free* - rozmiar wolnego bufora;
  - *pfree* - procent wolnego bufora;
  - *used* - rozmiar używanego bufora;
  - *pused* - procent używanego bufora.

Uwagi:

- Zwraca błąd „Proxy memory buffer is disabled”, gdy bufor pamięci jest wyłączony;
- Ta pozycja nie jest obsługiwana na serwerze Zabbix.

zabbix[proxy_buffer,state,changes]

 Zwraca liczbę zmian stanu między trybami bufora dysk/pamięć od momentu uruchomienia.
 Zwracane wartości: *Integer*; *0* - bufor pamięci jest wyłączony.

Uwagi:

- Częste zmiany stanu wskazują, że należy zwiększyć rozmiar lub wiek bufora pamięci.
- Jeśli stan bufora pamięci jest monitorowany rzadko (na przykład raz na minutę), bufor może zmienić swój stan bez zarejestrowania tego faktu.

zabbix[proxy_buffer,state,current]

 Zwraca bieżący stan działania, w którym przechowywane są nowe dane.
 Zwracane wartości: *0* - dysk; *1* - pamięć.

Komentarz:

- 0 jest również zwracane, gdy bufor pamięci jest wyłączony.

zabbix[proxy_history]

 Liczba wartości w tabeli historii proxy oczekujących na wysłanie do serwera.
 Zwracane wartości: *Integer*.

Komentarz:

- Ta pozycja nie jest obsługiwana na serwerze Zabbix.

zabbix[queue,<from>,<to>]

 Liczba monitorowanych pozycji w kolejce, które są opóźnione co najmniej o `<from>` sekund, ale mniej niż `<to>` sekund.
 Zwracana wartość: *Integer*.

Parametry:

- **from** - opóźnione co najmniej o (domyślnie 6 sekund);
- **to** - opóźnione co najwyżej o (domyślnie nieskończoność).

Komentarze:

- Time suffixes (`s,m,h,d,w`) są obsługiwane w parametrach.

Attention:
Zabbix proxy nie jest świadomy okresów konserwacji; szczegóły można znaleźć w sekcji Calculation of queues during maintenance.

zabbix[rcache,<cache>,<mode>]

 Statystyki dostępności pamięci podręcznej konfiguracji Zabbixa.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla procentu).

Parametry:

- **cache** - *buffer*;
- **mode**:
  - *total* - całkowity rozmiar bufora;
  - *free* - rozmiar wolnego bufora;
  - *pfree* - procent wolnego bufora;
  - *used* - rozmiar użytego bufora;
  - *pused* - procent użytego bufora.

zabbix[requiredperformance]

 Wymagana wydajność serwera Zabbix lub proxy Zabbix, wyrażona jako oczekiwana liczba nowych wartości na sekundę.
 Zwracana wartość: *Float*.

Komentarze:

- W przybliżeniu odpowiada pozycji "Required server performance, new values per second" w *Reports* > *System information*.

zabbix[stats,<ip>,<port>]

 Zwraca wewnętrzne metryki serwera Zabbix lub proxy. Jeśli zostaną określone `<ip>` i `<port>`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z lokalnej instancji.
 Zwracane wartości: *obiekt JSON*.

Parametry:

- **ip** - lista adresów IP/DNS/masek sieciowych serwerów/proxy, do których ma zostać wykonane zdalne zapytanie (domyślnie 127.0.0.1);
- **port** - port serwera/proxy, do którego ma zostać wykonane zdalne zapytanie (domyślnie 10051).

Uwagi:

- Żądanie stats będzie akceptowane tylko z adresów wymienionych w parametrze `StatsAllowedIP` server/proxy na docelowej instancji.
- Ten item zwraca wybrany zestaw wewnętrznych metryk. Szczegóły: zobacz Zdalne monitorowanie statystyk Zabbixa.

zabbix[stats,<ip>,<port>,queue,<from>,<to>]

 Zwraca wewnętrzne metryki kolejki (zobacz `zabbix[queue,<from>,<to>]`) serwera Zabbix lub proxy. Jeśli określono `<ip>` i `<port>`, metryki są pobierane ze zdalnej instancji; w przeciwnym razie z lokalnej instancji.
 Wartości zwracane: *obiekt JSON*.

Parametry:

- **ip** - lista adresów IP/DNS/masek sieciowych serwerów/proxy, które mają być zdalnie odpytywane (domyślnie `127.0.0.1`);
- **port** - port serwera/proxy, który ma być zdalnie odpytywany (domyślnie 10051);
- **from** - opóźnione co najmniej o (domyślnie 6 sekund);
- **to** - opóźnione co najwyżej o (domyślnie nieskończoność).

Uwagi:

- Żądanie statystyk będzie akceptowane tylko z adresów wymienionych w parametrze `StatsAllowedIP` serwer/proxy na docelowej instancji.
- Ten element zwraca wybrany zestaw wewnętrznych metryk. Szczegóły: zobacz Zdalne monitorowanie statystyk Zabbix.

Attention:
Proxy Zabbix nie jest świadomy okresów konserwacji; szczegóły: zobacz Obliczanie kolejek podczas konserwacji.

zabbix[tcache,cache,<parameter>]

 Statystyki efektywności pamięci podręcznej funkcji trendów Zabbix.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla procentu).

Parametry:

- **parameter**:
  - *all* - łączna liczba żądań do pamięci podręcznej (domyślnie);
  - *hits* - trafienia w pamięć podręczną;
  - *phits* - procent trafień w pamięć podręczną;
  - *misses* - chybienia pamięci podręcznej;
  - *pmisses* - procent chybionych odwołań do pamięci podręcznej;
  - *items* - liczba elementów w pamięci podręcznej;
  - *requests* - liczba żądań w pamięci podręcznej;
  - *pitems* - procent elementów w pamięci podręcznej względem elementów w pamięci podręcznej + żądań. Niski procent najprawdopodobniej oznacza, że rozmiar pamięci podręcznej można zmniejszyć.
  - *paccessed* - procent wpisów użytych w ciągu ostatnich 24 godzin.

Komentarze:

- Ta pozycja nie jest obsługiwana na proxy Zabbix.

zabbix[triggers]

 Liczba włączonych wyzwalaczy w bazie danych Zabbix, ze wszystkimi pozycjami włączonymi na włączonych hostach.
 Zwracana wartość: *Integer*.

Komentarze:

- Ta pozycja nie jest obsługiwana przez Zabbix proxy.

zabbix[uptime]

 Czas działania procesu serwera lub proxy Zabbix w sekundach.
 Zwracana wartość: *Integer*.

zabbix[vcache,buffer,<mode>]

 Statystyki dostępności pamięci podręcznej wartości Zabbix.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla procentu).

Parametry:

- **mode**:
  - *total* - całkowity rozmiar bufora;
  - *free* - rozmiar wolnego bufora;
  - *pfree* - procent wolnego bufora;
  - *used* - rozmiar używanego bufora;
  - *pused* - procent używanego bufora.

Komentarze:

- Ta pozycja nie jest obsługiwana na proxy Zabbix.

zabbix[vcache,cache,<parameter>]

 Statystyki skuteczności pamięci podręcznej wartości Zabbixa.
 Zwracana wartość: *Integer*. Jeśli <parameter> jest ustawiony na *mode*, zwracana wartość to: *0* - tryb normalny; *1* - tryb niskiej pamięci.

Parametry:

- **parameter**:
  - *requests* - łączna liczba żądań;
  - *hits* - liczba trafień w pamięć podręczną (wartości historii pobrane z pamięci podręcznej);
  - *misses* - liczba nietrafień w pamięć podręczną (wartości historii pobrane z bazy danych);
  - *mode* - tryb pracy pamięci podręcznej wartości.

Uwagi:

- Po włączeniu trybu niskiej pamięci pamięć podręczna wartości pozostanie w tym stanie przez 24 godziny, nawet jeśli problem, który wywołał ten tryb, zostanie rozwiązany wcześniej.

- Możesz użyć tego klucza z krokiem przetwarzania *Change per second*, aby uzyskać statystyki wartości na sekundę.
- Ta pozycja nie jest obsługiwana na proxy Zabbixa.

zabbixversion

 Wersja serwera lub proxy Zabbix.
 Zwracana wartość: *String*. Na przykład: `8.0.0`.

zabbix[vmware,buffer,<mode>]

 Statystyki dostępności pamięci podręcznej Zabbix vmware.
 Zwracane wartości: *Integer* (dla rozmiaru); *Float* (dla procentu).

Parametry:

- **mode**:
  - *total* - całkowity rozmiar bufora;
  - *free* - rozmiar wolnego bufora;
  - *pfree* - procent wolnego bufora;
  - *used* - rozmiar używanego bufora;
  - *pused* - procent używanego bufora.

zabbix[vps,written]

 Łączna liczba wartości historii zapisanych do bazy danych.
 Zwracana wartość: *Integer*.

zabbix[wcache,<cache>,<mode>]

 Statystyki i dostępność zapisu cache Zabbix.
 Zwracane wartości: *Integer* (dla liczby/rozmiaru); *Float* (dla procentu).

Parametry:

- **cache** - *values*, *history*, *index* lub *trend*;
- **mode**:
  - dla parametru cache *values*:
    - *all* - łączna liczba wartości przetworzonych przez serwer/proxy Zabbix, z wyjątkiem nieobsługiwanych pozycji (licznik) (domyślnie);
    - *float* - liczba przetworzonych wartości zmiennoprzecinkowych (licznik);
    - *uint* - liczba przetworzonych wartości całkowitych bez znaku (licznik);
    - *str* - liczba przetworzonych wartości znakowych/łańcuchowych (licznik);
    - *log* - liczba przetworzonych wartości logów (licznik);
    - *text* - liczba przetworzonych wartości tekstowych (licznik);
    - *bin* - liczba przetworzonych wartości binarnych (licznik);
    - *json* - liczba przetworzonych wartości JSON (licznik);
    - *not supported* - liczba przypadków, w których przetwarzanie pozycji spowodowało, że stała się ona nieobsługiwana lub pozostała w tym stanie (licznik);
  - dla parametrów cache *history*, *index*, *trend*:
    - *pfree* - procent wolnego bufora (domyślnie);
    - *total* - całkowity rozmiar bufora;
    - *free* - rozmiar wolnego bufora;
    - *used* - rozmiar użytego bufora;
    - *pused* - procent użytego bufora.

Uwagi:

- Parametr cache `trend` nie jest obsługiwany przez proxy Zabbix.
- Cache historii jest używany do przechowywania wartości pozycji. Niska wartość wskazuje na problemy wydajnościowe po stronie bazy danych.
- Cache indeksu historii jest używany do indeksowania wartości przechowywanych w cache historii.
- Po zapełnieniu cache historii i jego późniejszym wyczyszczeniu cache indeksu historii nadal będzie przechowywać część danych. Takie zachowanie jest oczekiwane i pomaga systemowi działać wydajniej, ponieważ unika dodatkowego przetwarzania wymaganego do ciągłego zmieniania rozmiaru pamięci.
- Cache trendów przechowuje agregat dla bieżącej godziny dla wszystkich pozycji, które otrzymują dane.
- Możesz użyć klucza `zabbix[wcache,values]` wraz z krokiem przetwarzania *Change per second*, aby uzyskać statystyki wartości na sekundę.

Zabbix trapper

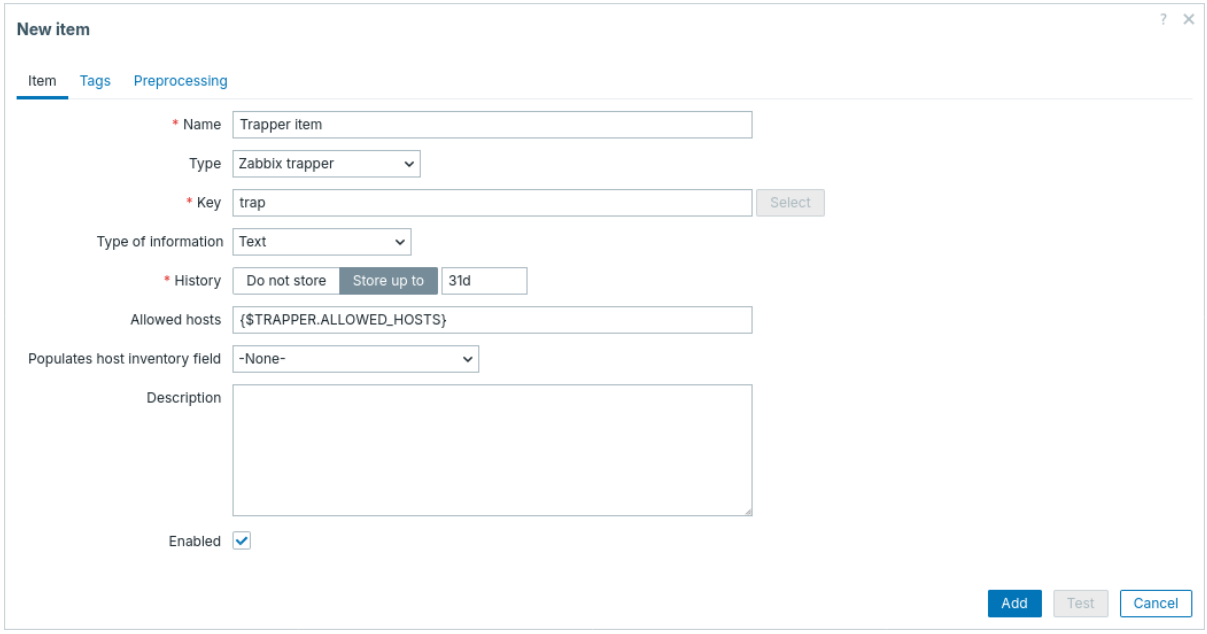
Przegląd

Pozycje trapper przyjmują dane przychodzące zamiast je odpytywać. Jest to przydatne w przypadku dowolnych danych, które chcesz wysłać do Zabbixa.

Konfiguracja

Aby skonfigurować pozycję typu trapper:

1. Przejdź do *Data collection* > *Hosts*.
2. Kliknij *Items* w wierszu hosta.
3. Kliknij *Create item*.
4. Wprowadź parametry pozycji w formularzu konfiguracji.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola, które wymagają określonych informacji dla pozycji typu trapper, to:

Type	Wybierz ”Zabbix trapper”.
Key	Wprowadź klucz, który będzie używany do rozpoznawania pozycji podczas wysyłania danych do serwer Zabbix.
Type of information	Wybierz typ informacji, który będzie odpowiadał formatowi wysyłanych danych.
Allowed hosts	Lista adresów IP rozdzielonych przecinkami (opcjonalnie w notacji CIDR) lub nazw DNS. Wartością domyślną jest makro użytkownika `{$TRAPPER.ALLOWED_HOSTS}`. Domyślnie `{$TRAPPER.ALLOWED_HOSTS}` ma wartość `127.0.0.1,::1`. Możesz wpisać adresy bezpośrednio w polu albo zmienić wartość makra. Jeśli pole pozostanie puste, połączenia przychodzące będą odrzucane ze wszystkich hostów. Jeśli zostanie określone, połączenia przychodzące będą akceptowane tylko od hostów wymienionych tutaj. Jeśli obsługa IPv6 jest włączona, to ’127.0.0.1’, ’::127.0.0.1’, ’::ffff:127.0.0.1’ są traktowane jednakowo, a ’::/0’ zezwoli na dowolny adres IPv4 lub IPv6. ’0.0.0.0/0’ można użyć, aby zezwolić na dowolny adres IPv4. Zwróć uwagę, że ”IPv4-compatible IPv6 addresses” (prefiks 0000::/96) są obsługiwane, ale zostały uznane za przestarzałe przez RFC4291. Przykład: 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} Obsługiwane są spacje, makra użytkownika oraz makra hosta {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN}.

Note:
Przed wysłaniem wartości może być konieczne odczekanie do 60 sekund po zapisaniu pozycji, aż serwer Zabbix pobierze zmiany z aktualizacji pamięci podręcznej konfiguracji.

Wysyłanie danych

Wysyłanie danych do serwera Zabbix lub proxy jest możliwe za pomocą narzędzia Zabbix sender lub protokołu Zabbix sender. Wysyłanie danych do serwera Zabbix jest również możliwe za pomocą metody API `history.push`.

Zabbix sender

Aby wysłać dane do serwera Zabbix lub proxy za pomocą narzędzia Zabbix sender, można uruchomić następujące polecenie, aby wysłać wartość "test value":

```
zabbix_sender -z <server IP address> -p 10051 -s "New host" -k trap -o "test value"
```

Aby wysłać wartość "test value", używane są następujące opcje polecenia:

- `-z` do określenia adresu IP serwera Zabbix
- `-p` do określenia numeru portu serwera Zabbix (domyślnie 10051)
- `-s` do określenia hosta (upewnij się, że używasz nazwy technicznej zamiast widocznej nazwy hosta)
- `-k` do określenia klucza pozycja skonfigurowanej w pozycji typu trapper
- `-o` do określenia wartości do wysłania

Attention:
Proces Zabbix trapper nie rozwija makr używanych w kluczu pozycji, aby sprawdzić istnienie odpowiedniego klucza pozycji dla docelowego hosta.

Więcej informacji o komunikacji między Zabbix sender a serwerem Zabbix lub proxy można znaleźć w sekcji *Zabbix sender protocol*.

history.push

Aby wysłać dane do serwera Zabbix za pomocą metody API `history.push`, można wykonać następujące żądanie HTTP POST zawierające przykładowe wartości testowe:

```
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":"test value 1"},{"ite
```

Jeśli żądanie jest poprawne, odpowiedź zwrócona przez API może wyglądać następująco:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "data": [
            {
                "itemid": "10600"
            },
            {
                "itemid": "10601",
                "error": "Item is disabled."
            },
            {
                "error": "No permissions to referred object or it does not exist."
            }
        ]
    },
    "id": 1
}
```

Błędy w danych odpowiedzi wskazują, że wysyłanie danych dla określonych pozycji nie przeszło walidacji przez serwer Zabbix. Może się tak zdarzyć z następujących powodów:

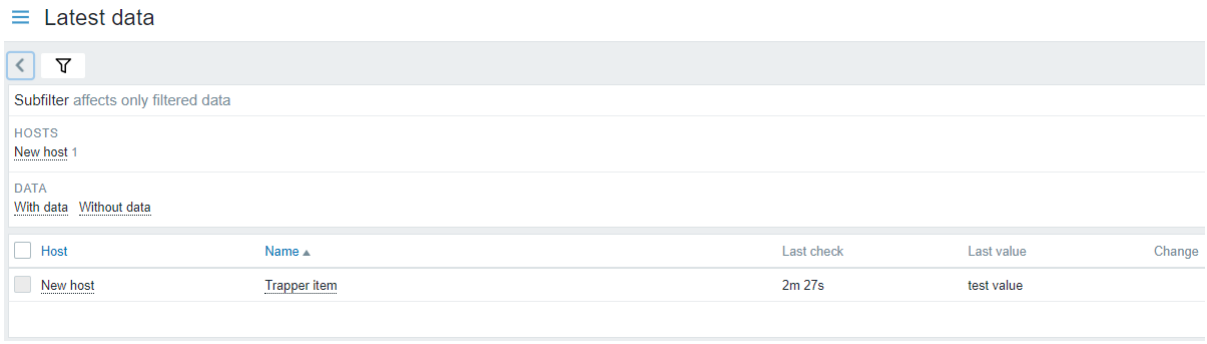
- użytkownik wysyłający dane nie ma uprawnień *read* do hosta danej pozycji;
- host jest wyłączony lub znajduje się w trybie maintenance bez zbierania danych;
- pozycja nie istnieje lub nie została jeszcze uwzględniona w pamięci podręcznej konfiguracji serwera;
- pozycja jest wyłączona lub jej typ jest inny niż Zabbix trapper lub HTTP agent (z włączonym trappingiem);
- adres IP lub DNS użytkownika nie jest ustawiony na liście *Allowed hosts* pozycji;
- inna pozycja ma wartość z duplikującym się znacznikiem czasu na poziomie nanosekund.

Brak błędów oznacza, że wysłane wartości zostały przyjęte do przetwarzania, które obejmuje preprocessing (jeśli występuje), przetwarzanie wyzwalaczy oraz zapis do bazy danych. Należy pamiętać, że przetwarzanie przyjętej wartości również może się nie powieść (na przykład podczas preprocessing), co spowoduje odrzucenie wartości.

Więcej informacji o pracy z Zabbix API można znaleźć w sekcji API.

Wyświetlanie danych

Po wysłaniu danych możesz przejść do *Monitoring* > *Latest data*, aby zobaczyć wynik:



Note:
Jeśli zostanie wysłana pojedyncza wartość liczbowa, wykres danych pokaże poziomą linię po lewej i prawej stronie punktu czasowego tej wartości.

Zewnętrzny check

Przegląd

Zewnętrzne sprawdzenie to sprawdzenie wykonywane przez serwer Zabbix przez uruchomienie skryptu powłoki lub pliku binarnego. Jednak gdy hosty są monitorowane przez proxy Zabbix, zewnętrzne sprawdzenia są wykonywane przez proxy.

Zewnętrzne sprawdzenia nie wymagają uruchomionego agenta na monitorowanym hoście.

Składnia klucza pozycji jest następująca:

```
script[<parameter1>,<parameter2>,...]
```

Gdzie:

ARGUMENT	DEFINICJA
script	Nazwa skryptu powłoki lub pliku binarnego.
parameter(s)	Opcjonalne parametry wiersza poleceń.

Jeśli nie chcesz przekazywać żadnych parametrów do skryptu, możesz użyć:

```
script[] or
script
```

Serwer Zabbix lub proxy wyszuka wskazany katalog dla skryptów zewnętrznych i wykona polecenie (zobacz parametr `ExternalScripts` w pliku konfiguracyjnym Zabbix serwer/proxy). Polecenie zostanie wykonane z uprawnieniami tego samego użytkownika, co serwer/proxy Zabbix, więc w razie potrzeby wszelkie uprawnienia dostępu lub zmienne środowiskowe powinny być obsłużone w skrypcie opakowującym. Uprawnienia do polecenia powinny również umożliwiać temu użytkownikowi jego wykonanie. Do wykonania dostępne są wyłącznie polecenia znajdujące się w określonym katalogu.

Warning:
Nie nadużywaj zewnętrznych sprawdzeń, ponieważ każdy skrypt wymaga uruchomienia procesu fork przez serwer/proxy Zabbix, a uruchamianie wielu skryptów może znacząco obniżyć wydajność Zabbix.

Przykład użycia

Wykonanie skryptu `check_oracle.sh` z pierwszym parametrem `'-h'`. Drugi parametr zostanie zastąpiony adresem IP lub nazwą DNS, w zależności od wyboru w właściwościach hosta.

```
check_oracle.sh["-h","{HOST.CONN}"]
```

Zakładając, że host jest skonfigurowany do używania adresu IP, serwer/proxy Zabbix wykona:

```
check_oracle.sh '-h' '192.168.1.4'
```

Wynik kontroli zewnętrznej

Wartością zwracaną kontroli zewnętrznej jest standardowe wyjście wraz ze standardowym wyjściem błędów generowanym przez kontrolę.

Attention:
Pozycja zwracająca tekst (informacja typu znakowego, log lub tekst) nie stanie się nieobsługiwana w przypadku standardowego wyjścia błędów.

Wartość zwracana jest ograniczona do 16 MB (włącznie z końcowymi białymi znakami, które są obcinane); obowiązują również ograniczenia bazy danych.

Jeśli żądany skrypt nie zostanie znaleziony lub serwer/proxy Zabbix nie ma uprawnień do jego wykonania, pozycja stanie się nieobsługiwana i zostanie wyświetlony odpowiedni komunikat o błędzie.

W przypadku przekroczenia limitu czasu pozycja stanie się nieobsługiwana, zostanie wyświetlony odpowiedni komunikat o błędzie, a proces utworzony dla skryptu zostanie zakończony.

Monitor bazy danych

Omówienie

Typ pozycja *Database monitor* w frontendzie Zabbix jest używany do monitorowania ODBC (sprawdzeń ODBC).

ODBC to pośredniczące API języka programowania C służące do uzyskiwania dostępu do systemów zarządzania bazami danych (DBMS). Koncepcja ODBC została opracowana przez firmę Microsoft, a następnie przeniesiona na inne platformy.

Zabbix może wykonywać zapytania do dowolnej bazy danych obsługiwanej przez ODBC. Aby to zrobić, Zabbix nie łączy się bezpośrednio z bazami danych, lecz korzysta z interfejsu ODBC oraz sterowników skonfigurowanych w ODBC. Umożliwia to bardziej efektywne monitorowanie różnych baz danych do wielu celów (na przykład sprawdzanie określonych kolejek bazy danych, statystyk użycia itp.).

Zabbix obsługuje unixODBC, który jest jedną z najczęściej używanych otwartoźródłowych implementacji API ODBC.

Attention:
Zobacz także: znane problemy dotyczące sprawdzeń ODBC.

Instalacja unixODBC

Zalecanym sposobem instalacji unixODBC jest użycie domyślnych repozytoriów pakietów systemu operacyjnego Linux. W najpopularniejszych dystrybucjach Linux unixODBC jest domyślnie uwzględniony w repozytorium pakietów. Jeśli pakiety nie są dostępne, pliki źródłowe można pobrać ze strony głównej unixODBC: http://www.unixodbc.org/download.html.

Aby zainstalować unixODBC, użyj menedżera pakietów dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install unixodbc unixodbc-dev

##### Dla systemów opartych na RedHat/Fedora:
dnf install unixODBC unixODBC-devel

##### Dla systemów opartych na SUSE:
zypper in unixODBC-devel
```

Attention:
Pakiet `unixodbc-dev` lub `unixODBC-devel` jest niezbędny do skompilowania Zabbixa z obsługą unixODBC. Aby włączyć obsługę ODBC, Zabbix należy skompilować z następującą opcją konfiguracji:

`--with-unixodbc[=ARG] # Użyj sterownika ODBC dla pakietu unixODBC.`

Instalowanie sterowników unixODBC

Sterownik bazy danych unixODBC powinien być zainstalowany dla bazy danych, która będzie monitorowana. Listę obsługiwanych baz danych i sterowników można znaleźć na stronie głównej unixODBC: http://www.unixodbc.org/drivers.html.

> **Note:**
> W niektórych dystrybucjach Linuksa sterowniki baz danych są dołączone do repozytoriów pakietów.

MySQL

Aby zainstalować sterownik bazy danych MySQL dla unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install odbc-mariadb

##### Dla systemów opartych na RedHat/Fedora:
dnf install mariadb-connector-odbc

##### Dla systemów opartych na SUSE:
zypper install mariadb-connector-odbc
```

Aby zainstalować sterownik bazy danych bez użycia menedżera pakietów, zapoznaj się z dokumentacją MySQL dla `mysql-connector-odbc` lub z dokumentacją MariaDB dla `mariadb-connector-odbc`.

PostgreSQL

Aby zainstalować sterownik bazy danych PostgreSQL dla unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install odbc-postgresql

##### Dla systemów opartych na RedHat/Fedora:
dnf install postgresql-odbc

##### Dla systemów opartych na SUSE:
zypper install psqlODBC
```

Aby zainstalować sterownik bazy danych bez użycia menedżera pakietów, zapoznaj się z dokumentacją PostgreSQL.

Oracle

Aby zainstalować sterownik bazy danych unixODBC, zapoznaj się z dokumentacją Oracle.

MSSQL

Aby zainstalować sterownik bazy danych MSSQL unixODBC, użyj menedżera pakietów odpowiedniego dla wybranego systemu:

```
##### Dla systemów Ubuntu/Debian:
apt install tdsodbc

##### Dla systemów opartych na RedHat/Fedora (pakiety EPEL: https://docs.fedoraproject.org/en-US/epel/):
dnf install epel-release
dnf install freetds

##### Dla systemów opartych na SUSE:
zypper install libtdsodbc0
```

Aby zainstalować sterownik bazy danych bez użycia menedżera pakietów, zapoznaj się z przewodnikiem użytkownika FreeTDS.

Konfigurowanie unixODBC

Aby skonfigurować unixODBC, należy edytować pliki `odbcinst.ini` i `odbc.ini`. Lokalizację tych plików można sprawdzić, wykonując następujące polecenie:

```
odbcinst -j
```

Wynik polecenia powinien zawierać informacje podobne do poniższych:

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

odbcinst.ini

Plik `odbcinst.ini` zawiera listę zainstalowanych sterowników baz danych ODBC. Jeśli plik `odbcinst.ini` nie istnieje, należy utworzyć go ręcznie.

```
[TEST_MYSQL]
Description=ODBC for MySQL
Driver=/usr/lib/libmyodbc5.so
FileUsage=1
```

Parameter	Description
TEST_MYSQL	Nazwa sterownika bazy danych.
Description	Opis sterownika bazy danych.
Driver	Lokalizacja biblioteki sterownika bazy danych.
FileUsage	Określa, czy sterownik bazy danych obsługuje połączenie z serwerem bazy danych bez obsługi dostępu do plików lokalnych (0); obsługuje odczyt danych z plików (1); obsługuje zapis danych do plików (2).
Threading	Poziom serializacji wątków. Obsługiwane dla PostgreSQL. Od wersji 1.6, jeśli menedżer sterowników został zbudowany z obsługą wątków, można dodać kolejny wpis sterownika.

odbc.ini

Plik `odbc.ini` służy do konfigurowania źródeł danych. Należy pamiętać, że lista obsługiwanych parametrów zależy od sterownika bazy danych (na przykład bazy danych Oracle mogą używać parametru ServerName zamiast Server itd.).

```
[TEST_MYSQL]
Description=MySQL Test Database
Driver=mysql
Server=127.0.0.1
User=root
Password=
Port=3306
Socket=
Database=zabbix
```

Parameter	Description
TEST_MYSQL	Nazwa źródła danych (DSN).
Description	Opis źródła danych.
Driver	Nazwa sterownika bazy danych (zgodnie z określeniem w `odbcinst.ini`).
Server	IP/DNS serwera bazy danych.
User	Użytkownik bazy danych używany do połączenia.
Password	Hasło użytkownika bazy danych.
Port	Port połączenia z bazą danych.
Socket	Gniazdo połączenia z bazą danych.
Database	Nazwa bazy danych.

Aby poznać inne możliwe opcje parametrów konfiguracyjnych, zobacz dokumentację MySQL.

Plik `odbc.ini` dla PostgreSQL może zawierać dodatkowe parametry:

```
[TEST_PSQL]
Description=PostgreSQL Test Database
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=
```

Parametr	Opis
ReadOnly	Określa, czy połączenie z bazą danych zezwala wyłącznie na operacje odczytu (zapytania SELECT) i ogranicza modyfikacje (instrukcje INSERT, UPDATE i DELETE); przydatne w scenariuszach, w których dane powinny pozostać niezmienione.
Protocol	Wersja protokołu backend PostgreSQL (ignorowana w przypadku używania połączeń SSL).
ShowOidColumn	Określa, czy uwzględniać Object ID (OID) w SQLColumns.
FakeOidIndex	Określa, czy tworzyć sztuczny unikalny indeks dla OID.
RowVersioning	Określa, czy włączyć aplikacjom możliwość wykrywania, czy dane zostały zmodyfikowane przez innych użytkowników podczas próby aktualizacji wiersza. Należy pamiętać, że ten parametr może przyspieszyć proces aktualizacji, ponieważ przy aktualizacji wiersza nie trzeba określać każdej pojedynczej kolumny w klauzuli WHERE.
ShowSystemTables	Określa, czy sterownik bazy danych powinien traktować tabele systemowe jako zwykłe tabele w SQLTables; jest to przydatne ze względu na dostępność, ponieważ umożliwia wgląd w tabele systemowe.
Fetch	Określa, czy sterownik powinien automatycznie używać declare cursor/fetch do obsługi instrukcji SELECT i utrzymywać pamięć podręczną 100 wierszy.
BoolsAsChar	Steruje mapowaniem typów logicznych. Jeśli ustawiono wartość "Yes", wartości logiczne są mapowane na SQL_CHAR; w przeciwnym razie są mapowane na SQL_BIT.
SSLmode	Określa tryb SSL dla połączenia.
ConnSettings	Dodatkowe ustawienia wysyłane do backend podczas nawiązywania połączenia.

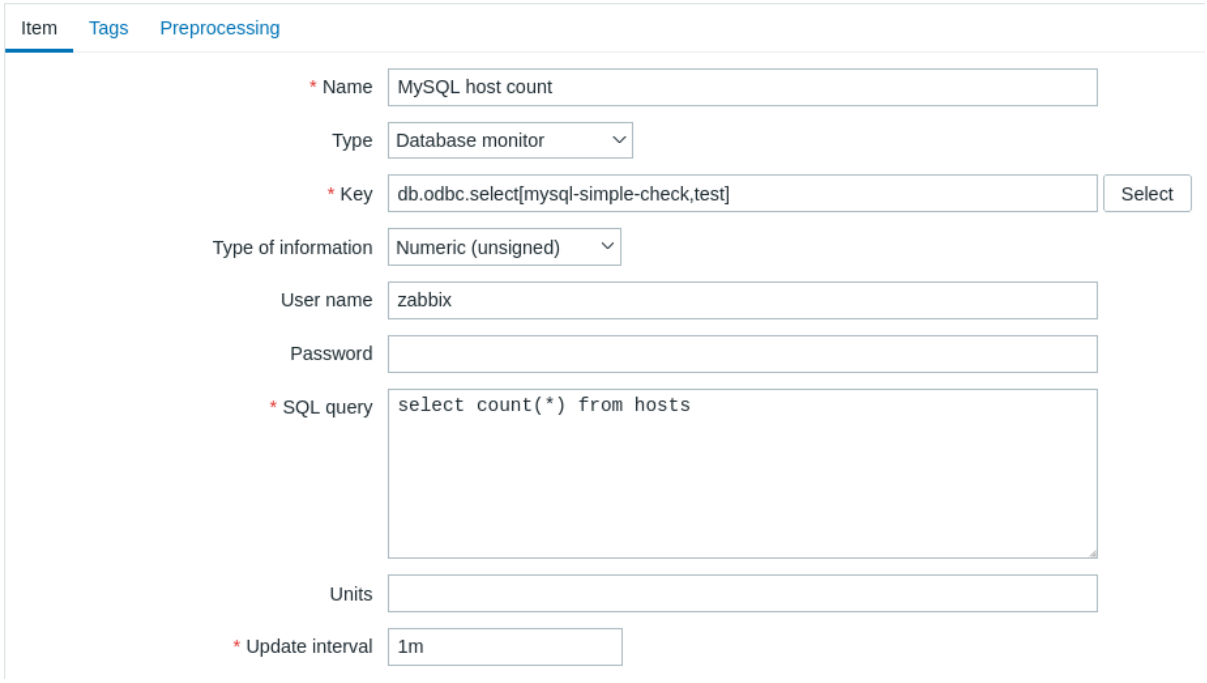
Testowanie połączenia ODBC

Aby sprawdzić, czy połączenie ODBC działa poprawnie, można użyć narzędzia `isql` (dołączonego do pakietu `unixODBC`):

```
isql test
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
|                                       |
+---------------------------------------+
```

Konfiguracja pozycji w frontend Zabbix

Skonfiguruj pozycję monitorowania **bazy danych**.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

W przypadku pozycji monitorowania bazy danych należy określić:

Type	Wybierz tutaj "Database monitor".
Key	Wprowadź jeden z obsługiwanych kluczy pozycji: **db.odbc.select[]** - ta pozycja zwraca jedną wartość (pierwszą kolumnę pierwszego wiersza wyniku zapytania SQL); **db.odbc.get[]** - ta pozycja zwraca wiele wierszy/kolumn w formacie JSON; **db.odbc.discovery[]** - ta pozycja zwraca dane wykrywania niskiego poziomu.
User name	Wprowadź nazwę użytkownika bazy danych (do 255 znaków). Ten parametr jest opcjonalny, jeśli nazwa użytkownika bazy danych jest określona w pliku `odbc.ini`. Jeśli używany jest ciąg połączenia, a pole *User name* nie jest puste, zostanie ono dołączone do ciągu połączenia jako `UID=<user>`.
Password	Wprowadź hasło użytkownika bazy danych (do 255 znaków). Ten parametr jest opcjonalny, jeśli hasło jest określone w pliku `odbc.ini`. Jeśli używany jest ciąg połączenia, a pole *Password* nie jest puste, zostanie ono dołączone do ciągu połączenia jako `PWD=<password>`. W tym polu obsługiwane są znaki specjalne. Dla wszystkich baz danych z wyjątkiem Oracle hasła zawierające `}` muszą być ujęte w nawiasy klamrowe `{}`. Hasło zostanie dołączone do ciągu połączenia po nazwie użytkownika, na przykład jako `UID=<username>;PWD=P?;)*word`. Aby przetestować wynikowy ciąg, możesz uruchomić następujące polecenie: `isql -v -k 'Driver=libmaodbc.so;Database=zabbix;UID=zabbix;PWD=P?;)*word'`
SQL query	Wprowadź zapytanie SQL. Pamiętaj, że w przypadku `db.odbc.select[]` zapytanie musi zwracać tylko jedną wartość.
Type of information	Wybierz tutaj typ informacji, który ma zostać zwrócony przez zapytanie. Jeśli typ informacji zostanie wybrany nieprawidłowo, pozycja stanie się nieobsługiwana.

Ważne uwagi

- Pozycje monitorowania bazy danych staną się nieobsługiwane, jeśli w konfiguracji serwera lub proxy nie zostaną uruchomione żadne procesy *odbc poller*. Aby aktywować ODBC pollers, ustaw parametr `StartODBCPollers` w pliku konfiguracyjnym Zabbix serwer lub, w przypadku kontroli wykonywanych przez proxy, w pliku konfiguracyjnym Zabbix proxy.
- Wartość parametru *Timeout* w formularzu konfiguracji pozycji jest używana jako limit czasu logowania ODBC oraz limit czasu wykonania zapytania. Należy pamiętać, że te ustawienia limitu czasu mogą zostać zignorowane, jeśli zainstalowany sterownik ODBC nie obsługuje ich.
- Polecenie SQL musi zwracać zestaw wyników, tak jak każde zapytanie używające instrukcji `select`. Składnia zapytania będzie zależeć od RDBMS, który będzie je przetwarzał. Składnia żądania do procedury składowanej musi zaczynać się od słowa kluczowego `call`.

Szczegóły klucza pozycji

Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

db.odbc.select[<unikalny krótki opis>,<dsn>,<connection string>]

 Zwraca jedną wartość, czyli pierwszą kolumnę pierwszego wiersza wyniku zapytania SQL.
 Wartość zwracana: Zależy od zapytania SQL.

Parametry:

- **unikalny krótki opis** — unikalny krótki opis identyfikujący pozycja (do użycia w wyzwalaczach itp.);
- **dsn** — nazwa źródła danych (zgodnie z definicją w `odbc.ini`);
- **connection string** — ciąg połączenia (może zawierać argumenty specyficzne dla sterownika).

Uwagi:

- Chociaż `dsn` i `connection string` są parametrami opcjonalnymi, wymagany jest co najmniej jeden z nich; jeśli zdefiniowano oba, `dsn` zostanie zignorowany.
- Jeśli zapytanie zwraca więcej niż jedną kolumnę, odczytywana jest tylko pierwsza kolumna. Jeśli zapytanie zwraca więcej niż jeden wiersz, odczytywany jest tylko pierwszy wiersz.

db.odbc.get[<unique short description>,<dsn>,<connection string>]

 Przekształca wynik zapytania SQL do tablicy JSON.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **unique short description** - unikalny krótki opis służący do identyfikacji pozycji (do użycia w wyzwalaczach itp.);
- **dsn** - nazwa źródła danych (zgodnie z definicją w `odbc.ini`);
- **connection string** - ciąg połączenia (może zawierać argumenty specyficzne dla sterownika).

Komentarze:

- Chociaż `dsn` i `connection string` są parametrami opcjonalnymi, wymagany jest co najmniej jeden z nich; jeśli oba są zdefiniowane, `dsn` zostanie zignorowany.
- W formacie JSON mogą zostać zwrócone wiele wierszy/kolumn. Ta pozycja może być używana jako pozycja nadrzędna, która zbiera wszystkie dane w jednym wywołaniu systemowym, natomiast w pozycjach zależnych można użyć przetwarzania JSONPath do wyodrębnienia poszczególnych wartości. Więcej informacji można znaleźć w przykładzie zwróconego formatu, używanego w niskopoziomowym wykrywaniu.

Przykład:

```
# 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[<unikalny krótki opis>,<dsn>,<connection string>]

 Przekształca wynik zapytania SQL w tablicę JSON, używaną do niskopoziomowego wykrywania. Nazwy kolumn z wyniku zapytania są przekształcane w nazwy makr niskopoziomowego wykrywania sparowane z wartościami wykrytych pól. Makra te mogą być używane podczas tworzenia prototypów pozycji, wyzwalaczy itp.
 Wartość zwracana: *obiekt JSON*.

Parametry:

- **unikalny krótki opis** — unikalny krótki opis identyfikujący pozycję (do użycia w wyzwalaczach itp.);
- **dsn** — nazwa źródła danych (zgodnie z definicją w `odbc.ini`);
- **connection string** — ciąg połączenia (może zawierać argumenty specyficzne dla sterownika).

Uwagi:

- Chociaż `dsn` i `connection string` są parametrami opcjonalnymi, wymagany jest co najmniej jeden z nich; jeśli zdefiniowano oba, `dsn` zostanie zignorowany.

Komunikaty o błędach

Komunikaty o błędach ODBC są podzielone na pola, aby dostarczyć szczegółowych informacji. Na przykład komunikat o błędzie może wyglądać tak:

```
Cannot execute ODBC query: [SQL_ERROR]:[42601][7][ERROR: syntax error at or near ";"; Error while executin
```

- "`Cannot execute ODBC query`" — komunikat Zabbix
- "`[SQL_ERROR]`" — kod zwrotny ODBC
- "`[42601]`" — SQLState
- "`[7]`" — natywny kod błędu
- "`[ERROR: syntax error at or near ";"; Error while executing the query]`" — natywny komunikat błędu

Należy pamiętać, że długość komunikatu o błędzie jest ograniczona do 2048 bajtów, więc komunikat może zostać obcięty. Jeśli istnieje więcej niż jeden rekord diagnostyczny ODBC, Zabbix próbuje je połączyć (oddzielając je znakiem |) w zakresie dozwolonym przez limit długości.

Agent HTTP

Przegląd

Ten typ pozycji umożliwia pobieranie danych przy użyciu protokołu HTTP/HTTPS. Możliwe jest również przyjmowanie danych za pomocą narzędzia Zabbix sender lub protokołu Zabbix sender (do wysyłania danych do serwera Zabbix lub proxy), albo przy użyciu metody API `history.push` (do wysyłania danych do serwera Zabbix).

Kontrole pozycji HTTP są wykonywane przez serwer Zabbix. Jednak gdy hosty są monitorowane przez proxy Zabbix, kontrole pozycji HTTP są wykonywane przez proxy.

Kontrole pozycji HTTP nie wymagają uruchomionego agenta na monitorowanym hoście.

HTTP agent obsługuje zarówno HTTP, jak i HTTPS. Zabbix opcjonalnie będzie podążać za przekierowaniami (zobacz opcję *Follow redirects* poniżej). Maksymalna liczba przekierowań jest na stałe ustawiona na 10 (z użyciem opcji cURL `CURLOPT_MAXREDIRS`).

> **Attention:**
> Serwer/proxy Zabbix musi być początkowo skonfigurowany z obsługą cURL (libcurl).

Kontrole HTTP są wykonywane asynchronicznie - nie jest wymagane otrzymanie odpowiedzi na jedno żądanie przed rozpoczęciem innych kontroli. Rozwiązywanie DNS również odbywa się asynchronicznie.

Maksymalna współbieżność asynchronicznych kontroli wynosi 1000 (zdefiniowana przez parametr MaxConcurrentChecksPerPoller).

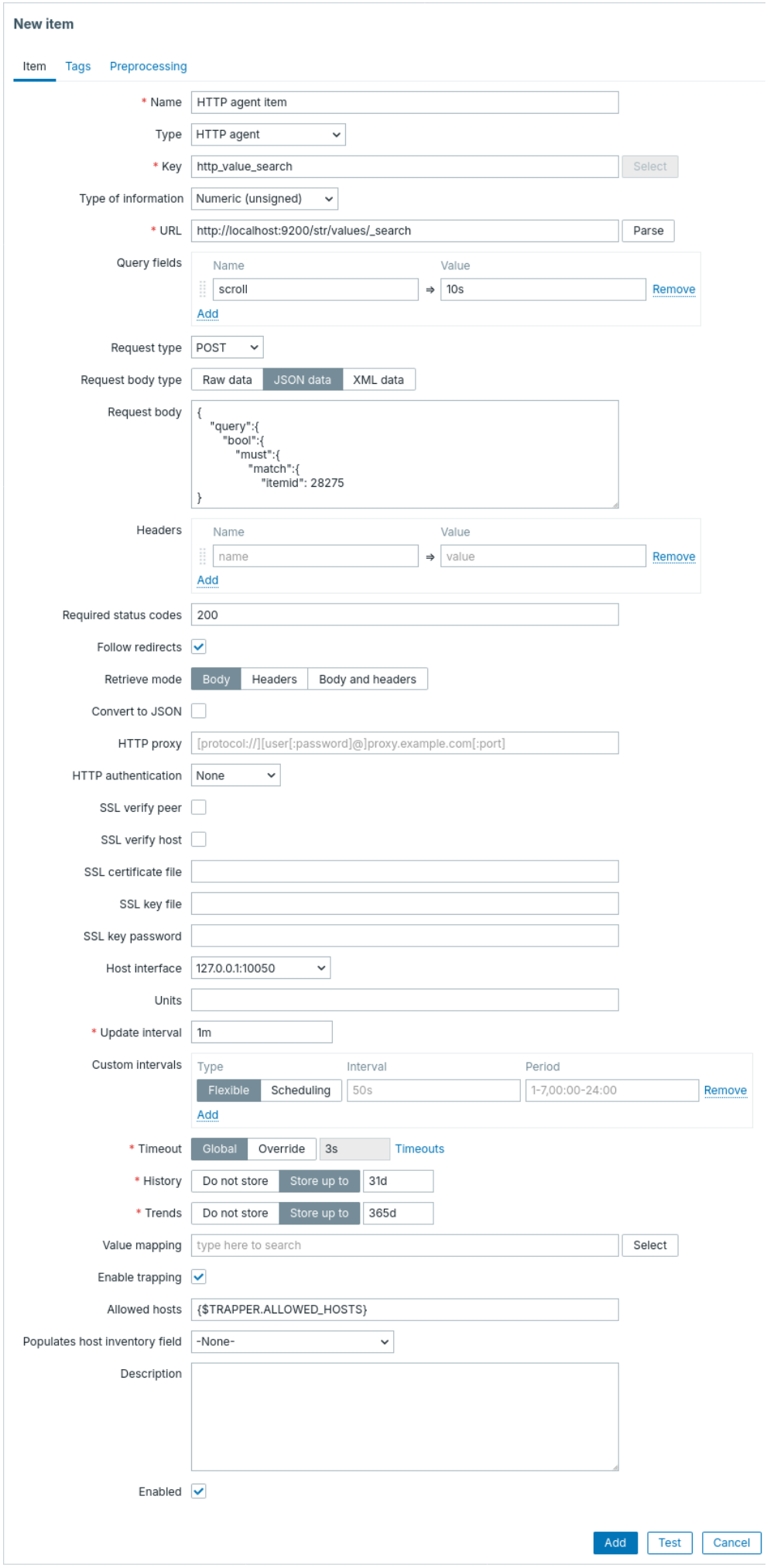
Liczba asynchronicznych pollerów HTTP agent jest określana przez parametr StartHTTPAgentPollers.

Funkcja trwałych połączeń cURL została dodana do kontroli HTTP agent od Zabbix 7.0.

Konfiguracja

Aby skonfigurować pozycję HTTP:

1. Przejdź do sekcji *Zbieranie danych* > *Hosty*.
2. Kliknij *Pozycje* w wierszu hosta.
3. Kliknij *Utwórz pozycję*.
4. Wprowadź parametry pozycji w formularzu.


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

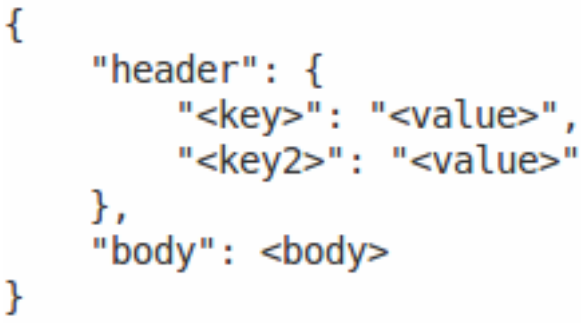
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji HTTP:

Parametr	Opis
Typ	Wybierz tutaj **HTTP agent**.
Klucz	Wprowadź unikalny klucz pozycji.

URL	URL, z którym należy się połączyć i pobrać dane. Na przykład:
	`https://www.example.com`
	`http://www.example.com/download`
	Nazwy domen mogą być określane za pomocą znaków Unicode. Podczas wykonywania sprawdzenia HTTP są one automatycznie konwertowane do ASCII w formacie punycode.
	Przycisk *Analizuj* umożliwia oddzielenie opcjonalnych pól zapytania (takich jak `?key=value&limit=10`) od URL, przenosząc atrybuty i wartości do pola *Pola zapytania* w celu automatycznego kodowania URL.
	Ze względów bezpieczeństwa nie umieszczaj rzeczywistych nazw użytkowników ani haseł w URL. Zamiast tego używaj makr użytkownika, aby uniknąć ujawnienia danych uwierzytelniających.
	Ograniczenie do 2048 znaków.
	Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra wykrywania niskiego poziomu.
	Ustawia opcję cURL CURLOPT_URL.

Pola zapytania	Zmienne dla URL (zobacz powyżej). Określane jako pary atrybutów i wartości. Wartości są automatycznie kodowane w URL. Wartości z makr są rozwijane, a następnie automatycznie kodowane w URL. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra wykrywania niskiego poziomu. Ustawia opcję cURL CURLOPT_URL.
Typ żądania	Wybierz typ metody żądania: *GET*, *POST*, *PUT* lub *HEAD*.
Typ treści żądania	Wybierz typ treści żądania: **Dane surowe** - niestandardowa treść żądania HTTP; makra są podstawiane, ale kodowanie nie jest wykonywane. **Dane JSON** - treść żądania HTTP w formacie JSON. Makra mogą być używane jako ciągi znaków, liczby, wartości true i false; makra używane jako ciągi znaków muszą być ujęte w podwójny cudzysłów. Wartości z makr są rozwijane, a następnie automatycznie zabezpieczane. Jeśli w nagłówkach nie określono „Content-Type", domyślnie zostanie użyte „Content-Type: application/json". **Dane XML** - treść żądania HTTP w formacie XML. Makra mogą być używane jako węzeł tekstowy, atrybut lub sekcja CDATA. Wartości z makr są rozwijane, a następnie automatycznie zabezpieczane w węźle tekstowym i atrybucie. Jeśli w nagłówkach nie określono „Content-Type", domyślnie zostanie użyte „Content-Type: application/xml".
Treść żądania	Wprowadź treść żądania. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra wykrywania niskiego poziomu.
Nagłówki	Niestandardowe nagłówki HTTP, które zostaną wysłane podczas wykonywania żądania. Określane jako pary atrybutów i wartości. Obsługiwane makra: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, makra użytkownika, makra wykrywania niskiego poziomu. Ustawia opcję cURL CURLOPT_HTTPHEADER.
Wymagane kody stanu	Lista oczekiwanych kodów stanu HTTP. Jeśli Zabbix otrzyma kod, którego nie ma na liście, pozycja stanie się nieobsługiwana. Jeśli lista jest pusta, sprawdzanie nie jest wykonywane. Na przykład: 200,201,210-299. Obsługiwane makra na liście: makra użytkownika, makra wykrywania niskiego poziomu. Używa opcji cURL CURLINFO_RESPONSE_CODE.
Podążaj za przekierowaniami	Zaznacz pole wyboru, aby podążać za przekierowaniami HTTP. Ustawia opcję cURL CURLOPT_FOLLOWLOCATION.

Parametr	Opis
Tryb pobierania	Wybierz część odpowiedzi, która ma zostać pobrana: **Treść** - tylko treść **Nagłówki** - tylko nagłówki **Treść i nagłówki** - treść i nagłówki
Konwertuj do JSON	Nagłówki są zapisywane jako pary atrybutów i wartości pod kluczem „header”, jeśli *Tryb pobierania* nie jest ustawiony na *Treść*. Jeśli zostanie napotkany „Content-Type: application/json”, treść jest zapisywana jako obiekt; w przeciwnym razie jest przechowywana jako ciąg znaków, na przykład: 
Proxy HTTP	Możesz określić proxy HTTP do użycia, korzystając z formatu `[protocol://][username[:password]@]proxy.example.com[:port]`. Opcjonalny prefiks `protocol://` może służyć do określenia alternatywnych protokołów proxy (np. `https`, `socks4`, `socks5`; zobacz dokumentację; obsługa prefiksu protokołu została dodana w cURL 7.21.7). Jeśli protokół nie zostanie określony, proxy będzie traktowane jako proxy HTTP. Jeśli określisz nieprawidłowy protokół, połączenie nie powiedzie się, a pozycja stanie się nieobsługiwana. Domyślnie używany będzie port 1080. Jeśli proxy zostanie określone, zastąpi powiązane z proxy zmienne środowiskowe, takie jak `http_proxy`, `HTTPS_PROXY`. Jeśli nie zostanie określone, nie zastąpi zmiennych środowiskowych związanych z proxy. Wprowadzona wartość jest przekazywana „w niezmienionej postaci”; nie jest wykonywane sprawdzanie poprawności. Należy pamiętać, że w przypadku proxy HTTP obsługiwane jest tylko uwierzytelnianie podstawowe. Obsługiwane makra: `{HOST.IP}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.PORT}`, `{ITEM.ID}`, `{ITEM.KEY}`, `{ITEM.KEY.ORIG}`, makra użytkownika, makra wykrywania niskiego poziomu. Ustawia opcję cURL `CURLOPT_PROXY`.
Uwierzytelnianie HTTP	Wybierz opcję uwierzytelniania: **Brak** - uwierzytelnianie nie jest używane; **Basic** - używane jest uwierzytelnianie podstawowe; **NTLM** - używane jest uwierzytelnianie NTLM (Windows NT LAN Manager); **Kerberos** - używane jest uwierzytelnianie Kerberos (zobacz także: Konfigurowanie Kerberos z Zabbix); **Digest** - używane jest uwierzytelnianie Digest. Ustawia opcję cURL `CURLOPT_HTTPAUTH`.
Nazwa użytkownika	Wprowadź nazwę użytkownika (do 255 znaków). To pole jest dostępne, jeśli *Uwierzytelnianie HTTP* jest ustawione na *Basic*, *NTLM*, *Kerberos* lub *Digest*. Obsługiwane są makra użytkownika i makra wykrywania niskiego poziomu.
Hasło	Wprowadź hasło użytkownika (do 255 znaków). To pole jest dostępne, jeśli *Uwierzytelnianie HTTP* jest ustawione na *Basic*, *NTLM*, *Kerberos* lub *Digest*. Obsługiwane są makra użytkownika i makra wykrywania niskiego poziomu.
Weryfikuj peer SSL	Zaznacz pole wyboru, aby zweryfikować certyfikat SSL serwera web. Certyfikat serwera jest weryfikowany względem certyfikatów zaufanych urzędów certyfikacji (CA) z systemowej lokalizacji CA. Lokalizację plików CA można zastąpić za pomocą parametru konfiguracji serwera Zabbix lub proxy SSLCALocation. Ustawia opcję cURL `CURLOPT_SSL_VERIFYPEER`.
Weryfikuj host SSL	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub *Subject Alternate Name* certyfikatu serwera web jest zgodne. Ustawia opcję cURL `CURLOPT_SSL_VERIFYHOST`.

Parametr	Opis
Plik certyfikatu SSL	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM[1]. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole *Plik klucza SSL* puste. Jeśli klucz jest zaszyfrowany, określ hasło w polu *Hasło klucza SSL*. Katalog zawierający ten plik jest określany przez parametr konfiguracji serwera Zabbix lub proxy `SSLCertLocation`. Obsługiwane makra: `{HOST.IP}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.PORT}`, `{ITEM.ID}`, `{ITEM.KEY}`, `{ITEM.KEY.ORIG}`, makra użytkownika, makra wykrywania niskiego poziomu. Ustawia opcję cURL `CURLOPT_SSLCERT`.
Plik klucza SSL	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik klucza prywatnego musi być w formacie PEM[1]. Katalog zawierający ten plik jest określany przez parametr konfiguracji serwera Zabbix lub proxy `SSLKeyLocation`. Obsługiwane makra: `{HOST.IP}`, `{HOST.CONN}`, `{HOST.DNS}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.PORT}`, `{ITEM.ID}`, `{ITEM.KEY}`, `{ITEM.KEY.ORIG}`, makra użytkownika, makra wykrywania niskiego poziomu. Ustawia opcję cURL `CURLOPT_SSLKEY`.
Hasło klucza SSL	Hasło pliku prywatnego klucza SSL. Obsługiwane makra: makra użytkownika, makra wykrywania niskiego poziomu. Ustawia opcję cURL `CURLOPT_KEYPASSWD`.
Limit czasu	Zabbix nie poświęci więcej czasu na przetwarzanie URL niż określona wartość (1-600 sekund). W rzeczywistości ten parametr określa maksymalny czas nawiązania połączenia z URL oraz maksymalny czas wykonania żądania HTTP. Dlatego Zabbix nie poświęci na jedno sprawdzenie więcej niż 2 x *Limit czasu* sekund. Ustawia opcję cURL `CURLOPT_TIMEOUT`. Więcej informacji o parametrze *Limit czasu* zawiera sekcja ogólne atrybuty pozycji.
Włącz przechwytywanie	Po zaznaczeniu tego pola wyboru pozycja będzie również działać jako pozycja trappera i będzie akceptować dane wysyłane do serwera Zabbix lub proxy za pomocą narzędzia Zabbix sender lub protokołu Zabbix sender, a także dane wysyłane do serwera Zabbix za pomocą metody API `history.push`. Więcej informacji o wysyłaniu danych zawiera sekcja: Pozycje trappera.
Dozwolone hosty	Widoczne tylko wtedy, gdy zaznaczono pole wyboru *Włącz przechwytywanie*. Lista rozdzielanych przecinkami adresów IP (opcjonalnie w notacji CIDR) lub nazw DNS. Wartością domyślną jest makro użytkownika `{$TRAPPER.ALLOWED_HOSTS}`. Domyślnie `{$TRAPPER.ALLOWED_HOSTS}` ma wartość `127.0.0.1,::1`. Możesz wprowadzić adresy bezpośrednio w polu lub zmienić wartość makra. Jeśli pole pozostanie puste, połączenia przychodzące ze wszystkich hostów będą odrzucane. Jeśli zostanie określone, połączenia przychodzące będą akceptowane tylko z wymienionych tutaj hostów. Jeśli obsługa IPv6 jest włączona, `127.0.0.1`, `::127.0.0.1`, `::ffff:127.0.0.1` są traktowane jednakowo, a `::/0` zezwoli na dowolny adres IPv4 lub IPv6. `0.0.0.0/0` może służyć do zezwolenia na dowolny adres IPv4. Należy pamiętać, że „adresy IPv6 kompatybilne z IPv4” (prefiks `0000::/96`) są obsługiwane, ale uznane za przestarzałe przez RFC4291. Przykład: `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}` W tym polu dozwolone są spacje i makra użytkownika. W tym polu dozwolone są makra hosta: `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.IP}`, `{HOST.DNS}`, `{HOST.CONN}`.

Note:
Jeśli pole *Proxy HTTP* pozostanie puste, innym sposobem użycia proxy HTTP jest ustawienie zmiennych środowiskowych związanych z proxy.
W przypadku HTTP ustaw zmienną środowiskową `http_proxy` dla użytkownika serwera Zabbix. Na przykład:
`http_proxy=http://proxy_ip:proxy_port`.
W przypadku HTTPS ustaw zmienną środowiskową `HTTPS_PROXY`. Na przykład:
`HTTPS_PROXY=http://proxy_ip:proxy_port`. Więcej informacji można uzyskać, uruchamiając polecenie powłoki: *# man curl*.

Attention:
[1] Zabbix obsługuje wyłącznie pliki certyfikatów i kluczy prywatnych w formacie PEM. Jeśli dane certyfikatu i klucza prywatnego znajdują się w pliku w formacie PKCS #12 (zwykle z rozszerzeniem `*.p12` lub `*.pfx`), możesz wygenerować z niego plik PEM za pomocą następujących poleceń:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes  -out ssl-cert.key
```

Przykłady

Przykład 1

Wysyłaj proste żądania GET, aby pobierać dane z usług takich jak Elasticsearch:

- Utwórz pozycję GET z adresem URL: `localhost:9200/?pretty`
- Zwróć uwagę na odpowiedź:

```
{
  "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" : "You know, for search"
}
```

- Teraz wyodrębnij numer wersji, używając kroku przetwarzania JSONPath: `$.version.number`

Przykład 2

Wyślij proste żądania POST, aby pobrać dane z usług takich jak Elasticsearch:

- Utwórz pozycję POST z adresem URL: `http://localhost:9200/str/values/_search?scroll=10s`
- Skonfiguruj następujące ciało POST, aby uzyskać obciążenie procesora (średnia z 1 minuty na rdzeń)

```
{
    "query": {
        "bool": {
            "must": [{
                "match": {
                    "itemid": 28275
                }
            }],
            "filter": [{
                "range": {
                    "clock": {
                        "gt": 1517565836,
                        "lte": 1517566137
                    }
                }
            }]
        }
    }
}
```

- Otrzymano:

```
{
    "_scroll_id": "DnF1ZXJ5VGhlbkZldGNoBQAAAAAAAAAkFllRMlZBWS1UU1pxTmdEeGVwQjRBTFEAAAAAAAAAJRZZUTJWQVk
    "took": 18,
    "timed_out": false,
```

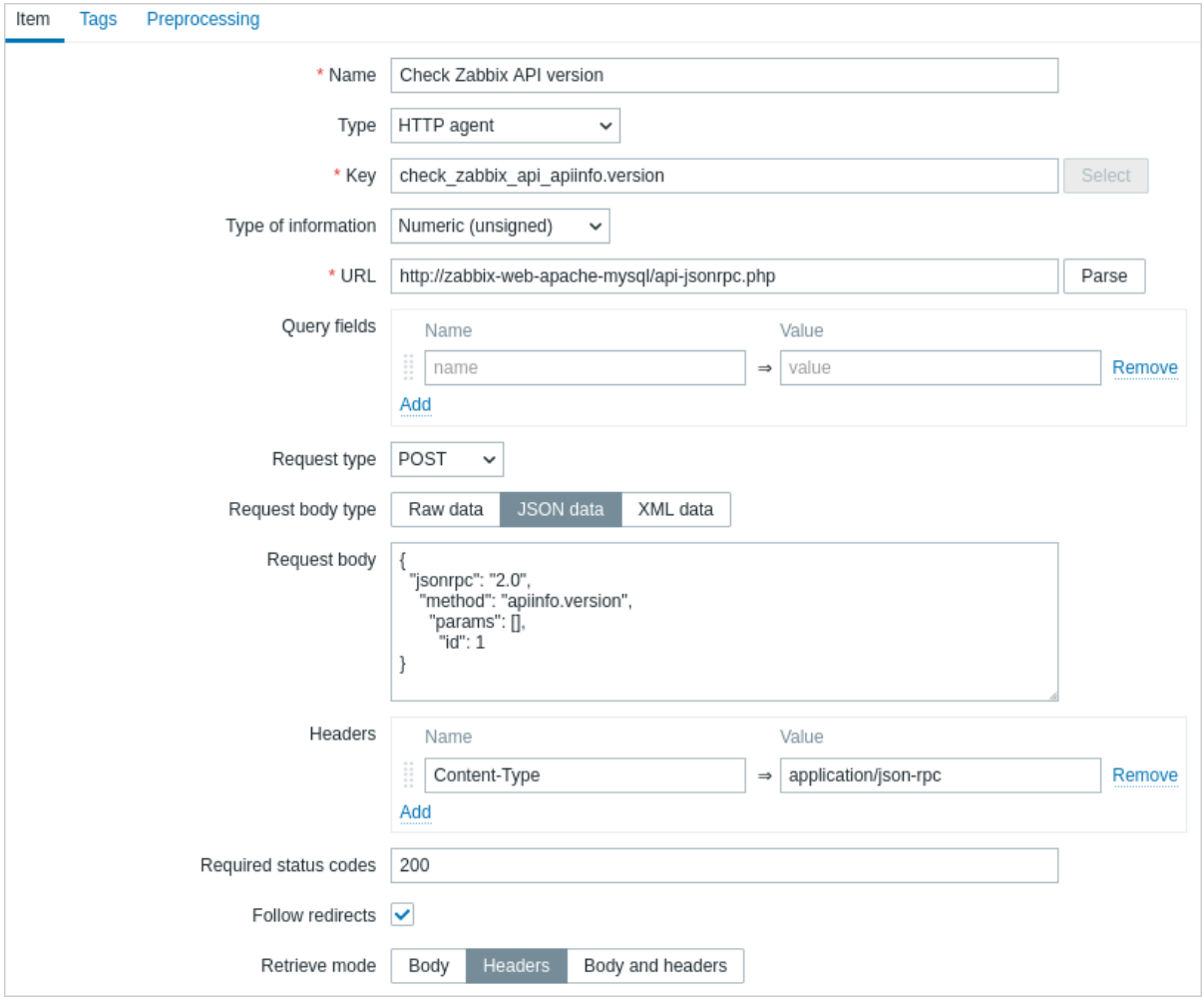
```
        "_shards": {
            "total": 5,
            "successful": 5,
            "skipped": 0,
            "failed": 0
        },
        "hits": {
            "total": 1,
            "max_score": 1.0,
            "hits": [{
                "_index": "dbl",
                "_type": "values",
                "_id": "dqX9VWEBV6sEKSMyk6sw",
                "_score": 1.0,
                "_source": {
                    "itemid": 28275,
                    "value": "0.138750",
                    "clock": 1517566136,
                    "ns": 25388713,
                    "ttl": 604800
                }
            }]
        }
    }
```


- Teraz użyj kroku przetwarzania JSONPath, aby pobrać wartość pozycji: `$.hits.hits[0]._source.value`

Przykład 3

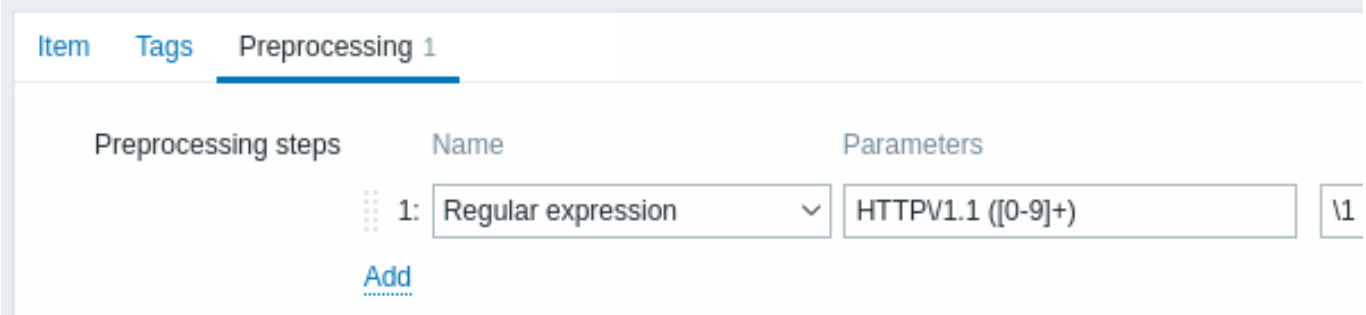
Sprawdzanie, czy Zabbix API działa, z użyciem apiinfo.version.

- Konfiguracja pozycji:

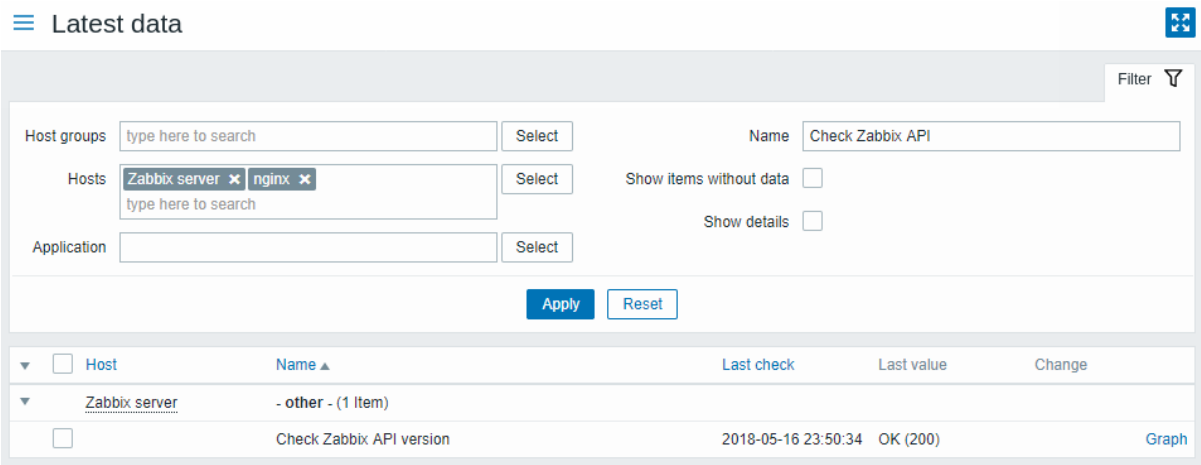


Zwróć uwagę na użycie metody POST z danymi JSON, ustawienie nagłówków żądania oraz żądanie zwrócenia tylko nagłówków:

- Wstępne przetwarzanie wartości pozycji za pomocą wyrażenia regularnego w celu uzyskania kodu HTTP:



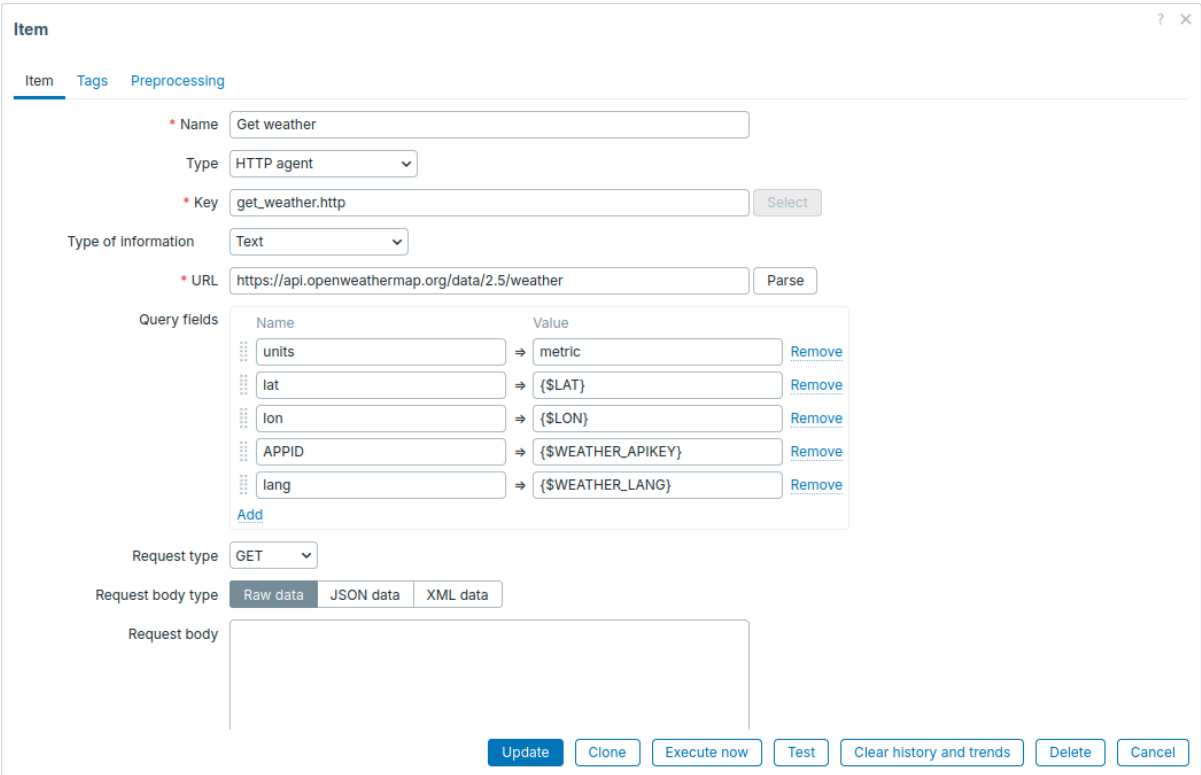
- Sprawdzanie wyniku w *Latest data*:



Przykład 4

Pobieranie informacji o pogodzie przez połączenie z publiczną usługą Openweathermap.

- Skonfiguruj pozycję nadrzędną do zbiorczego pobierania danych w jednym JSON:



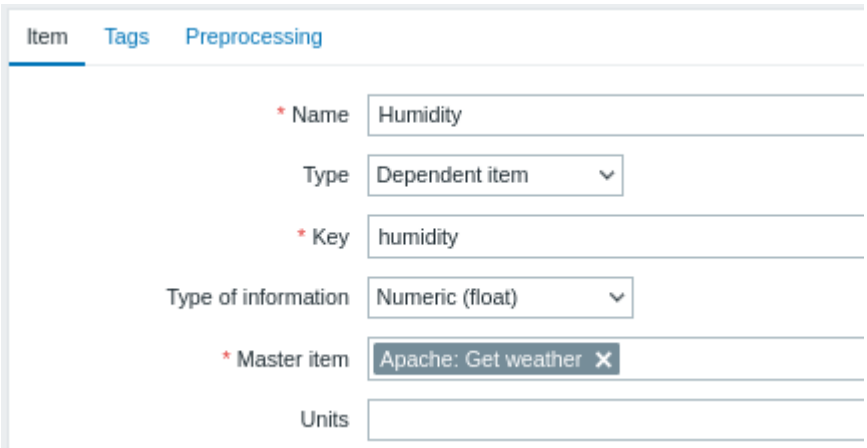
Zwróć uwagę na użycie makr w polach zapytania. Informacje o tym, jak je wypełnić, znajdziesz w Openweathermap API.

Przykładowy JSON zwracany w odpowiedzi przez HTTP agent:

```
{
    "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
    }
}
```

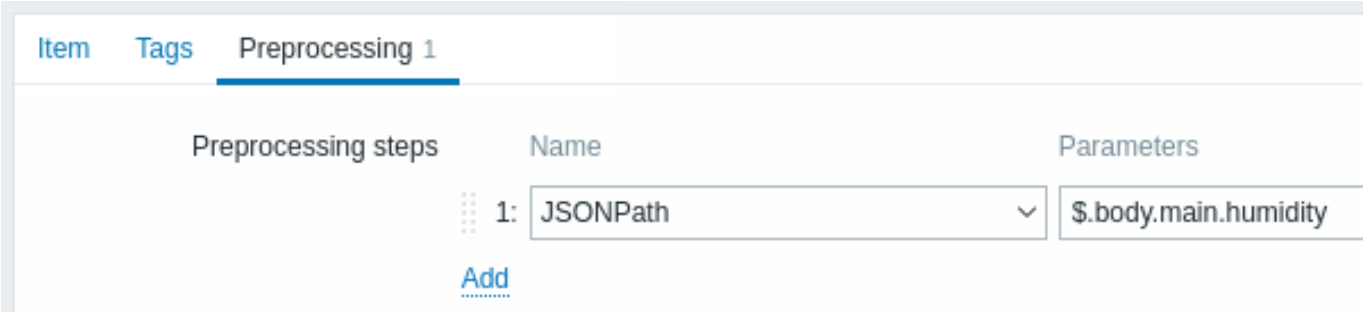

Następnym zadaniem jest skonfigurowanie pozycji zależnych, które wyodrębniają dane z JSON.

- Skonfiguruj przykładową pozycję zależną dla wilgotności:

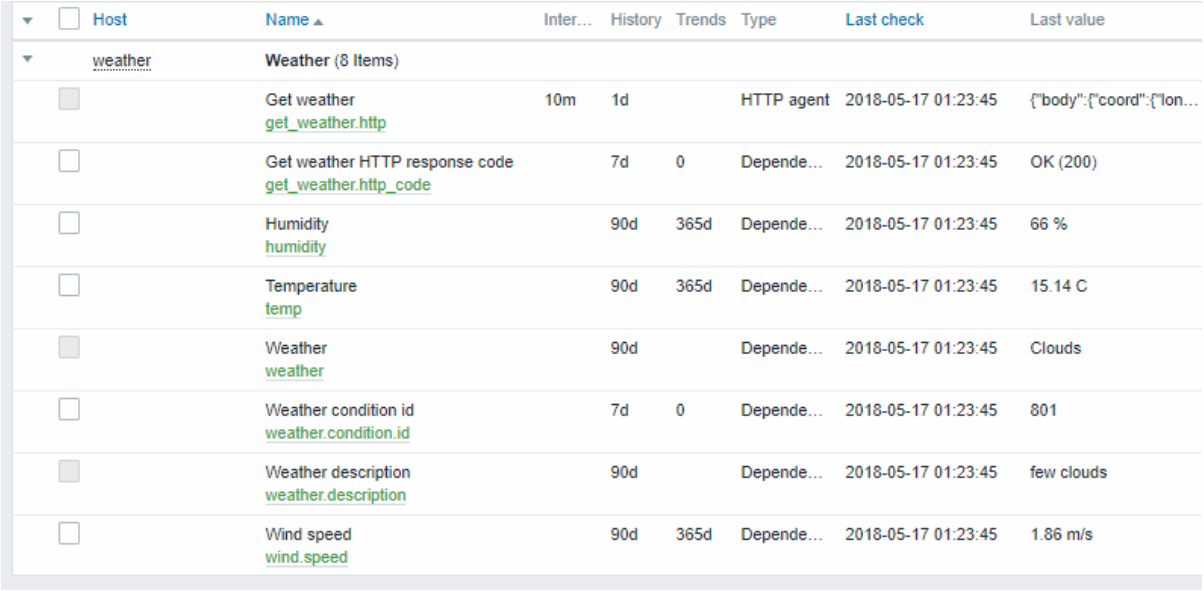


Pozostałe metryki pogodowe, takie jak „Temperatura”, dodaje się w ten sam sposób.

- Przykładowe przetwarzanie wstępne wartości pozycji zależnej z użyciem JSONPath:



- Sprawdź wynik danych pogodowych w *Latest data*:

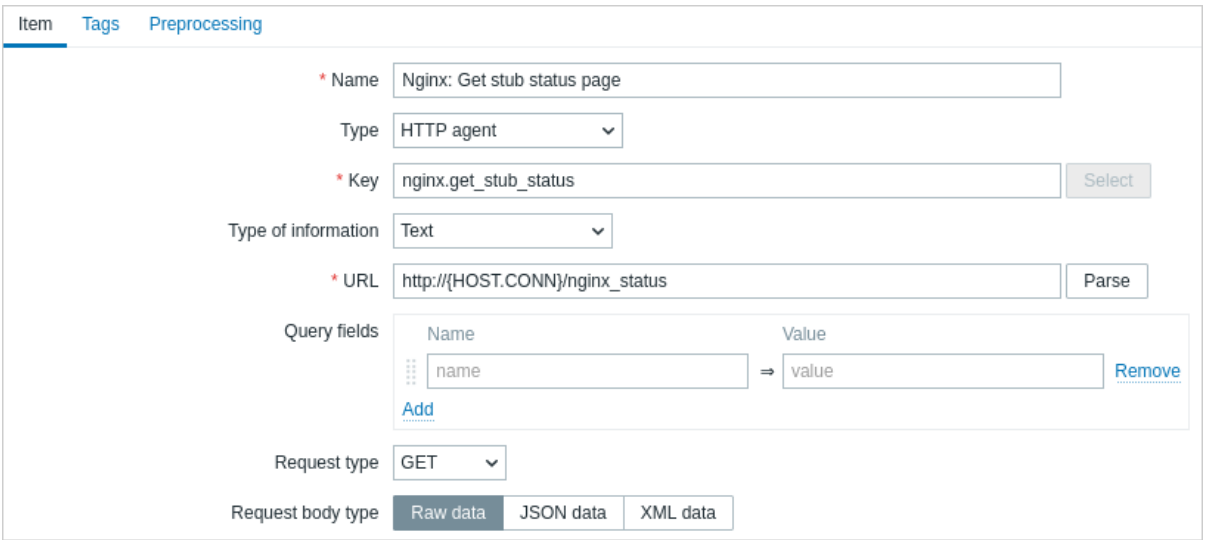


Host	Name	Inter...	History	Trends	Type	Last check	Last value
weather	**Weather** (8 Items)						
	Get weather get_weather.http	10m	1d		HTTP agent	2018-05-17 01:23:45	{"body":{"coord":{"lon...
	Get weather HTTP response code get_weather.http_code		7d	0	Depende...	2018-05-17 01:23:45	OK (200)
	Humidity 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

Przykład 5

Łączenie ze stroną stanu NGINX i pobieranie jego metryk zbiorczo.

- Skonfiguruj NGINX zgodnie z oficjalnym przewodnikiem.
- Skonfiguruj nadrzędną pozycję do zbierania danych zbiorczych:

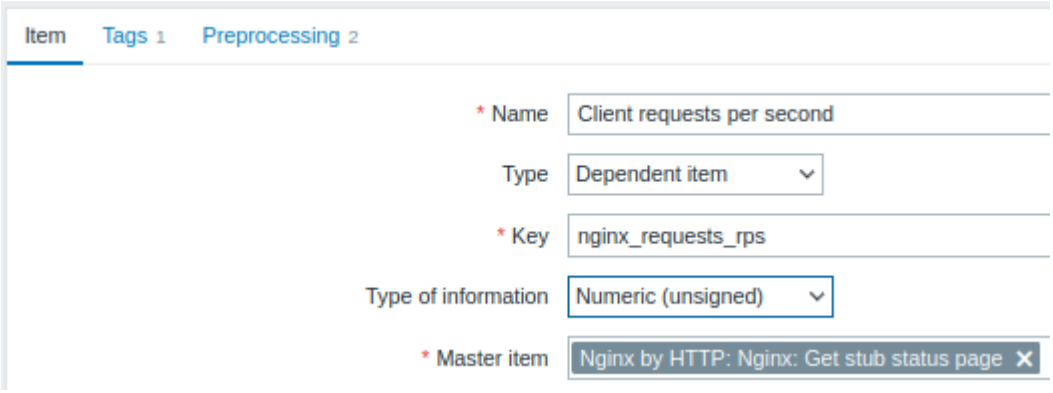


Przykładowe dane wyjściowe NGINX stub status:

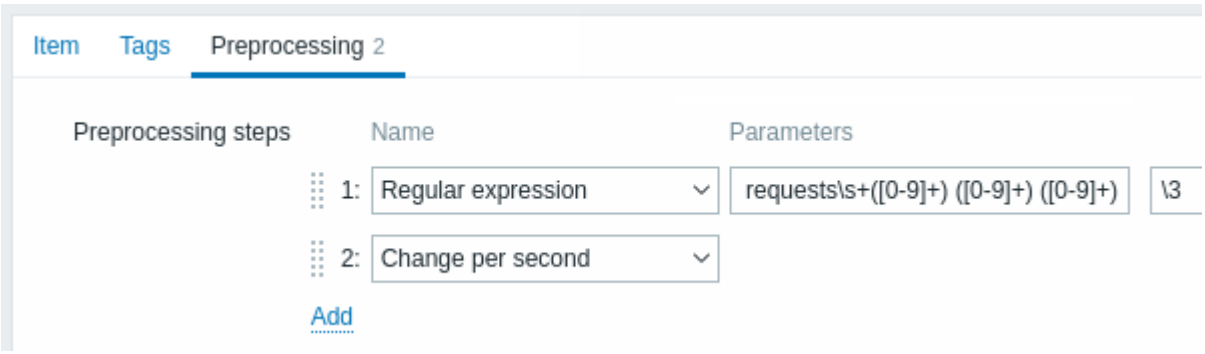
```
Active connections: 1 Active connections:
server accepts handled requests
 52 52 52
Reading: 0 Writing: 1 Waiting: 0
```

Następnym zadaniem jest skonfigurowanie pozycji zależnych, które wyodrębniają dane.

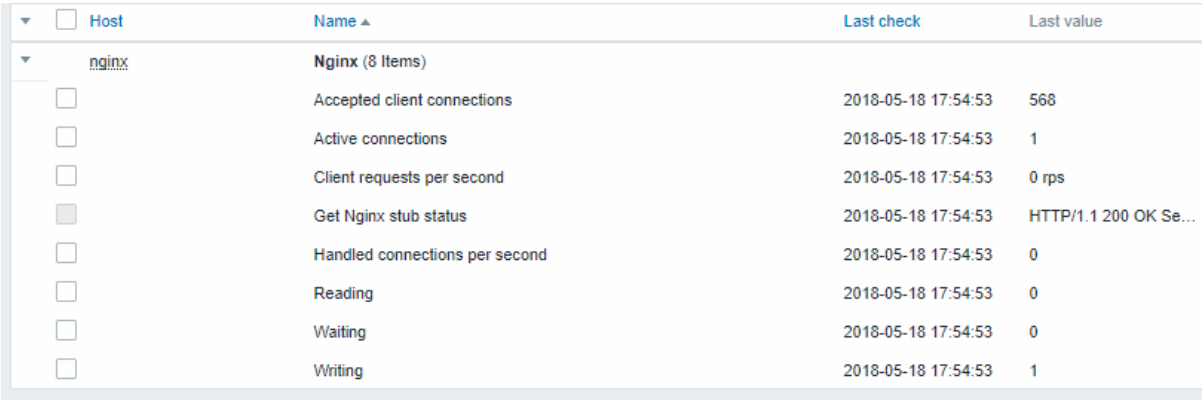
- Skonfiguruj przykładową pozycję zależną dla liczby żądań na sekundę:



- Przykładowe wstępne przetwarzanie wartości pozycji zależnej za pomocą wyrażenia regularnego `server accepts handled requests\s+([0-9]+) ([0-9]+) ([0-9]+)`:



- Sprawdź kompletny wynik z modułu stub w sekcji *Najnowsze dane*:



Sprawdzenie Prometheusa

Przegląd

Zabbix może odpytywać metryki udostępniane w formacie liniowym Prometheus.

Aby rozpocząć zbieranie danych Prometheus, wymagane są dwa kroki:

- główna pozycja HTTP wskazująca na odpowiedni punkt końcowy danych, np. `https://<prometheus host>/metrics`
- pozycje zależne korzystające z opcji przetwarzania wstępnego Prometheus do odpytywania wymaganych danych z metryk zebranych przez główną pozycję

Istnieją dwie opcje przetwarzania wstępnego danych Prometheus:

- *Wzorzec Prometheus* - używany w zwykłych pozycjach do odpytywania danych Prometheus
- *Prometheus do JSON* - używany w zwykłych pozycjach oraz do wykrywania niskiego poziomu. W tym przypadku odpytywane dane Prometheus są zwracane w formacie JSON.

Przetwarzanie zbiorcze

Przetwarzanie zbiorcze jest obsługiwane dla pozycji zależnych. Aby włączyć buforowanie i indeksowanie, preprocessing *Prometheus pattern* musi być **pierwszym** krokiem preprocessing. Gdy *Prometheus pattern* jest pierwszym krokiem preprocessing, przeanalizowane dane Prometheus są buforowane i indeksowane przez pierwszy warunek `<label>==<value>` w kroku

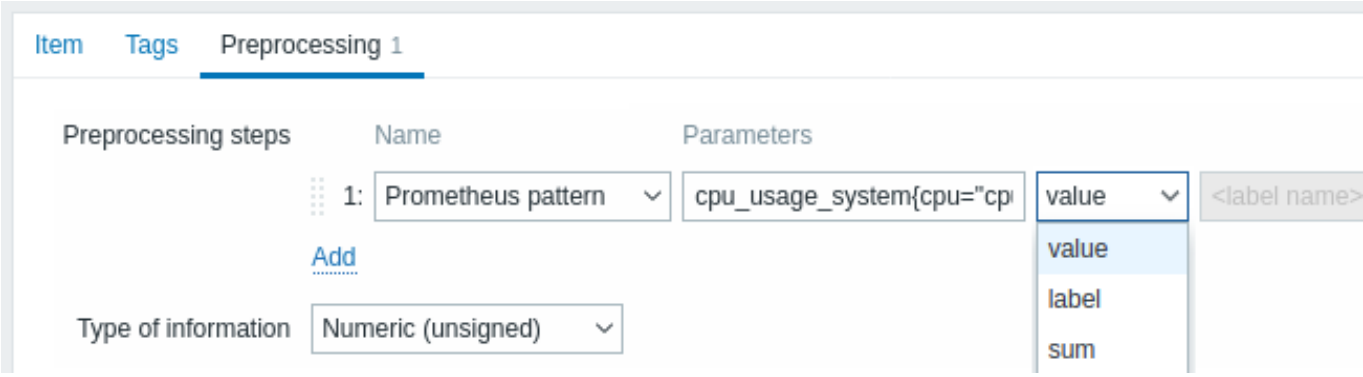
preprocessing *Prometheus pattern*. Ta pamięć podręczna jest ponownie używana podczas przetwarzania innych pozycji zależnych w tym wsadzie. Aby uzyskać optymalną wydajność, pierwsza etykieta powinna być tą, która ma najwięcej różnych wartości.

Jeśli przed pierwszym krokiem trzeba wykonać inne preprocessing, należy go przenieść albo do pozycji głównej, albo do nowej pozycji zależnej, która będzie używana jako pozycja główna dla pozycji zależnych.

Konfiguracja

Jeśli skonfigurowano HTTP master item, należy utworzyć pozycję zależną, która używa kroku przetwarzania wstępnego Prometheus:

1. Wprowadź ogólne parametry pozycji zależnej w formularzu konfiguracji.
2. Przejdź do karty *Preprocessing*.
3. Wybierz opcję przetwarzania wstępnego Prometheus (*Prometheus pattern* lub *Prometheus to JSON*).



Poniższe parametry są specyficzne dla opcji przetwarzania wstępnego *Prometheus pattern*:

Parameter	Description	Examples
Pattern	Aby zdefiniować wymagany wzorzec danych, można użyć języka zapytań podobnego do języka zapytań Prometheus (zobacz tabelę porównawczą), np.: <metric name> - wybór według nazwy metryki {__name__=”<metric name>”} - wybór według nazwy metryki {__name__=~”<regex>”} - wybór według nazwy metryki pasującej do wyrażenia regularnego {<label name>=”<label value>”,...} - wybór według nazwy etykiety {<label name>=~”<regex>”,...} - wybór według nazwy etykiety pasującej do wyrażenia regularnego {__name__=~”.*”}==<value> - wybór według wartości metryki Lub kombinację powyższych elementów: <metric name>{<label1 name>=”<label1 value>”,<label2 name>=~”<regex>”,...}==<value> Wartość etykiety może być dowolnym ciągiem znaków UTF-8, ale znaki odwrotnego ukośnika, cudzysłowu i końca wiersza muszą być poprzedzone znakami ucieczki jako \\, \" i \n; pozostałe znaki nie powinny być poprzedzane znakami ucieczki.	*wmi_os_physical_memory_free_bytes* *cpu_usage_system{cpu=”cpu-total”}* *cpu_usage_system{cpu=”.*”}* *cpu_usage_system{cpu=”.*”}* *wmi_service_state{name=”dhcp”}==1* *wmi_os_timezone{timezone=~”.*”}==1*

Parameter	Description	Examples
Result processing	Określ, czy zwrócić wartość, etykietę, czy zastosować odpowiednią funkcję (jeśli wzorzec pasuje do kilku wierszy i wynik wymaga agregacji): **value** - zwraca wartość metryki (błąd, jeśli pasuje wiele wierszy) **label** - zwraca wartość etykiety określonej w polu *Label* (błąd, jeśli pasuje wiele metryk) **sum** - zwraca sumę wartości **min** - zwraca wartość minimalną **max** - zwraca wartość maksymalną **avg** - zwraca wartość średnią **count** - zwraca liczbę wartości To pole jest dostępne tylko dla opcji *Prometheus pattern*.	Zobacz także poniższe przykłady użycia parametrów.
Output	Zdefiniuj nazwę etykiety (opcjonalnie). W takim przypadku zwracana jest wartość odpowiadająca nazwie etykiety. To pole jest dostępne tylko dla opcji *Prometheus pattern*, jeśli w polu *Result processing* wybrano opcję „Label”.	

Przykłady użycia parametrów

1. Najczęstszym przypadkiem użycia jest zwrócenie **wartości**. Aby zwrócić wartość `/var/db` z:

   `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`

użyj następujących parametrów:

- *Pattern* - `node_disk_usage_bytes{path="/var/db"}`
- *Result processing* - wybierz „value”

2. Możesz także potrzebować **średniej** wartości wszystkich parametrów `node_disk_usage_bytes`:

- *Pattern* - `node_disk_usage_bytes`
- *Result processing* - wybierz „avg”

3. Chociaż Prometheus obsługuje tylko dane liczbowe, często stosuje się obejście umożliwiające zwrócenie również odpowiedniego opisu tekstowego. Można to osiągnąć za pomocą filtra i określenia etykiety. Aby zwrócić wartość etykiety „color” z:

   `elasticsearch_cluster_health_status{cluster="elasticsearch",color="green"} 1<br> elasticsearch_cluster 0<br> elasticsearch_cluster_health_status{cluster="elasticsearch",color="yellow"} 0`

użyj następujących parametrów:

- *Pattern* - `elasticsearch_cluster_health_status {cluster="elasticsearch"} == 1`
- *Result processing* - wybierz „label”
- *Label* - określ „color”

Filtr (oparty na wartości liczbowej „1”) dopasuje odpowiedni wiersz, natomiast etykieta zwróci opis stanu kondycji (obecnie „green”, ale potencjalnie także „red” lub „yellow”).

Prometheus do JSON

Dane z Prometheus mogą być używane do niskopoziomowego wykrywania. W takim przypadku potrzebne są dane w formacie JSON, a opcja wstępnego przetwarzania *Prometheus to JSON* zwróci dokładnie taki wynik.

Więcej informacji można znaleźć w sekcji Wykrywanie z użyciem danych Prometheus.

Porównanie języka zapytań

Poniższa tabela przedstawia różnice i podobieństwa między PromQL a językiem zapytań preprocesora Zabbix Prometheus.

Selektor wektora chwilowego PromQL	Preprocessing Zabbix Prometheus

Różnice

Selektor wektora chwilowego PromQL		Preprocessing Zabbix Prometheus
Cel zapytania	serwer Prometheus	Zwykły tekst w formacie ekspozycji Prometheus
Zwraca	Wektor chwilowy	Wartość metryki lub etykiety (wzorzec Prometheus) Tablica metryk dla pojedynczej wartości w JSON (Prometheus to JSON)
Operatory dopasowania etykiet	**=**, **!=**, **=~**, **!~**	**=**, **!=**, **=~**, **!~**
Wyrażenie regularne używane w dopasowywaniu nazwy etykiety lub metryki	RE2	PCRE
Operatory porównania	Zobacz listę	Do filtrowania wartości obsługiwany jest tylko **==** (równe)
Podobieństwa		
Wybieranie według nazwy metryki równej ciągowi	<nazwa metryki> lub {__name__="<nazwa metryki>"}	<nazwa metryki> lub {__name__="<nazwa metryki>"}
Wybieranie według nazwy metryki pasującej do wyrażenia regularnego	{__name__=~"<regex>"}	{__name__=~"<regex>"}
Wybieranie według wartości <nazwa etykiety> równej ciągowi	{<nazwa etykiety>="<wartość etykiety>",...}	{<nazwa etykiety>="<wartość etykiety>",...}
Wybieranie według wartości <nazwa etykiety> pasującej do wyrażenia regularnego	{<nazwa etykiety>=~"<regex>",...}	{<nazwa etykiety>=~"<regex>",...}
Wybieranie według wartości równej ciągowi	{__name__=~".*"} == <wartość>	{__name__=~".*"} == <wartość>

Agent IPMI

Przegląd

W Zabbix można monitorować stan i dostępność urządzeń Intelligent Platform Management Interface (IPMI). Aby wykonywać kontrole IPMI, serwer Zabbix musi zostać początkowo skonfigurowany z obsługą IPMI.

IPMI to ustandaryzowany interfejs do zdalnego zarządzania systemami komputerowymi w trybie „lights-out" lub „out-of-band". Umożliwia monitorowanie stanu sprzętu bezpośrednio z tak zwanych kart zarządzania „out-of-band", niezależnie od systemu operacyjnego i od tego, czy urządzenie jest w ogóle włączone.

Monitorowanie IPMI w Zabbix działa tylko w przypadku urządzeń obsługujących IPMI (HP iLO, DELL DRAC, IBM RSA, Sun SSP itd.).

Proces menedżera IPMI planuje kontrole IPMI wykonywane przez pollery IPMI. W danym momencie host jest odpytywany zawsze tylko przez jeden poller IPMI, co zmniejsza liczbę otwartych połączeń z kontrolerami BMC. Dzięki temu można bezpiecznie zwiększyć liczbę pollerów IPMI bez obaw o przeciążenie kontrolera BMC. Proces menedżera IPMI jest uruchamiany automatycznie po uruchomieniu co najmniej jednego pollera IPMI.

Zobacz także znane problemy dotyczące kontroli IPMI.

Konfiguracja

Konfiguracja hosta

Aby przetwarzać kontrole IPMI, host musi być skonfigurowany. Należy dodać interfejs IPMI z odpowiednim adresem IP i numerem portu oraz zdefiniować parametry uwierzytelniania IPMI.

Więcej informacji można znaleźć w sekcji konfiguracja hostów.

Konfiguracja serwera

Domyślnie serwer Zabbix nie jest skonfigurowany do uruchamiania żadnych pollerów IPMI, więc wszystkie dodane pozycje IPMI nie będą działać. Aby to zmienić, otwórz plik konfiguracyjny serwera Zabbix (`zabbix_server.conf`) jako root i znajdź następującą linię:

```
# StartIPMIPollers=0
```

Usuń znak komentarza i ustaw liczbę pollerów na przykład na 3, tak aby linia wyglądała następująco:

```
StartIPMIPollers=3
```

Zapisz plik, a następnie uruchom ponownie zabbix_server.

Konfiguracja pozycji

Podczas konfigurowania pozycji na poziomie hosta:

- Wybierz „IPMI agent” jako *Typ*
- Wprowadź klucz pozycji, który jest unikalny w obrębie hosta (na przykład `ipmi.fan.rpm`)
- W polu *Interfejs hosta* wybierz odpowiedni interfejs IPMI (adres IP i port). Pamiętaj, że interfejs IPMI musi istnieć na hoście.
- Określ *Czujnik IPMI* (na przykład „FAN MOD 1A RPM” w Dell Poweredge), z którego ma być pobierana metryka. Domyślnie należy podać identyfikator czujnika. Możliwe jest również użycie prefiksów przed wartością:
  - `id:` — aby określić identyfikator czujnika;
  - `name:` — aby określić pełną nazwę czujnika. Może to być przydatne w sytuacjach, gdy czujniki można rozróżnić tylko przez podanie pełnej nazwy.
- Wybierz odpowiedni typ informacji („Numeryczny (zmiennoprzecinkowy)” w tym przypadku; dla czujników dyskretnych — „Numeryczny (bez znaku)”), jednostki (najprawdopodobniej „rpm”) oraz wszelkie inne wymagane atrybuty pozycji

Obsługiwane testy

agent IPMI obsługuje wbudowaną pozycję **ipmi.get**, która zwraca informacje związane z czujnikami IPMI i może być używana do wykrywania czujników IPMI.
 Wartość zwracana: *obiekt JSON*

Timeout i zakończenie sesji

Timeouty komunikatów IPMI oraz liczba ponowień są zdefiniowane w bibliotece OpenIPMI. Ze względu na obecną konstrukcję OpenIPMI nie ma możliwości skonfigurowania tych wartości w Zabbix, ani na poziomie interfejsu, ani pozycja.

Timeout bezczynności sesji IPMI dla LAN wynosi 60 +/-3 sekundy. Obecnie nie jest możliwe zaimplementowanie okresowego wysyłania polecenia Activate Session przy użyciu OpenIPMI. Jeśli z Zabbix do określonego BMC nie są wykonywane żadne kontrole pozycja IPMI przez czas dłuższy niż timeout sesji skonfigurowany w BMC, to następna kontrola IPMI po upływie tego timeoutu zakończy się timeoutem z powodu timeoutów pojedynczych komunikatów, ponowień lub błędu odbioru. Następnie otwierana jest nowa sesja i inicjowane jest pełne ponowne skanowanie BMC. Może zostać otwarty nowy port UDP do zarządzania nową sesją. Bezczynność jest definiowana jako brak zarówno żądań wychodzących, jak i odpowiedzi przychodzących. Jeśli chcesz uniknąć niepotrzebnych ponownych skanowań BMC, zaleca się ustawienie interwału odpytywania pozycja IPMI poniżej timeoutu bezczynności sesji IPMI skonfigurowanego w BMC.

Uwagi dotyczące dyskretnych sensorów IPMI

Aby znaleźć sensory na hoście, uruchom serwer Zabbix z włączonym **DebugLevel=4**. Odczekaj kilka minut i znajdź rekordy wykrywania sensorów w pliku logu serwera 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
```

Do dekodowania typów i stanów sensorów IPMI dostępna jest kopia specyfikacji IPMI 2.0 (proszę zauważyć, że nie są planowane dalsze aktualizacje specyfikacji IPMI).

Pierwszym parametrem, od którego należy zacząć, jest ”reading_type”. Użyj ”Table 42-1, Event/Reading Type Code Ranges” ze specyfikacji, aby zdekodować kod ”reading_type”. Większość sensorów w naszym przykładzie ma ”reading_type:0x1”, co oznacza sensor typu ”threshold”. ”Table 42-3, Sensor Type Codes” pokazuje, że ”type:0x1” oznacza sensor temperatury, ”type:0x2” —

sensor napięcia, "type:0x4" — wentylator itd. Sensory progowe są czasami nazywane sensorami "analogowymi", ponieważ mierzą ciągłe parametry, takie jak temperatura, napięcie, obroty na minutę.

Inny przykład — sensor z "reading_type:0x3". "Table 42-1, Event/Reading Type Code Ranges" mówi, że kody typu odczytu 02h-0Ch oznaczają sensor "Generic Discrete". Sensory dyskretne mają do 15 możliwych stanów (innymi słowy — do 15 znaczących bitów). Na przykład dla sensora 'CATERR' z "type:0x7" "Table 42-3, Sensor Type Codes" pokazuje, że ten typ oznacza "Processor", a znaczenie poszczególnych bitów jest następujące: 00h (bit najmniej znaczący) — IERR, 01h — Thermal Trip itd.

W naszym przykładzie jest kilka sensorów z "reading_type:0x6f". Dla tych sensorów "Table 42-1, Event/Reading Type Code Ranges" zaleca użycie "Table 42-3, Sensor Type Codes" do dekodowania znaczenia bitów. Na przykład sensor 'Power Unit Stat' ma typ "type:0x9", co oznacza "Power Unit". Przesunięcie 00h oznacza "PowerOff/Power Down". Innymi słowy, jeśli bit najmniej znaczący ma wartość 1, to serwer jest wyłączony. Aby sprawdzić ten bit, można użyć funkcji `bitand` z maską '1'. Wyrażenie wyzwalacza może wyglądać następująco:

```
bitand(last(/www.example.com/Power Unit Stat,#1),1)=1
```

aby ostrzegać o wyłączeniu zasilania serwera.

Uwagi dotyczące nazw czujników dyskretnych w OpenIPMI-2.0.16, 2.0.17, 2.0.18 i 2.0.19

Nazwy czujników dyskretnych w OpenIPMI-2.0.16, 2.0.17 i 2.0.18 często mają na końcu dodane dodatkowe "0" (lub inną cyfrę albo literę). Na przykład, podczas gdy `ipmitool` i OpenIPMI-2.0.19 wyświetlają nazwy czujników jako "`PhysicalSecurity`" lub "`CATERR`", w OpenIPMI-2.0.16, 2.0.17 i 2.0.18 nazwy te to odpowiednio "`PhysicalSecurity0`" lub "`CATERR0`".

Podczas konfigurowania pozycji IPMI w serwerze Zabbix z użyciem OpenIPMI-2.0.16, 2.0.17 i 2.0.18 należy używać tych nazw zakończonych na "0" w polu *IPMI sensor* pozycji agenta IPMI. Gdy serwer Zabbix zostanie zaktualizowany do nowej dystrybucji Linuksa, która używa OpenIPMI-2.0.19 (lub nowszej), pozycje z tymi dyskretnymi czujnikami IPMI staną się "NOT SUPPORTED". Należy zmienić ich nazwy *IPMI sensor* (usunąć końcowe '0') i odczekać pewien czas, aż ponownie staną się "Enabled".

Uwagi dotyczące jednoczesnej dostępności czujnika progowego i czujnika dyskretnego

Niektóre agenty IPMI udostępniają zarówno czujnik progowy, jak i czujnik dyskretny pod tą samą nazwą. Zawsze preferowany jest czujnik progowy.

Uwagi dotyczące zakończenia połączenia

Jeśli kontrole IPMI nie są wykonywane (z dowolnego powodu: wszystkie pozycje IPMI hosta są wyłączone/nieobsługiwane, host jest wyłączony/usunięty, host jest w trybie konserwacji itp.), połączenie IPMI zostanie zakończone przez serwer Zabbix lub proxy w ciągu 3 do 4 godzin, w zależności od czasu uruchomienia serwera/proxy Zabbix.

SSH agent

Omówienie

Kontrole SSH są wykonywane jako monitorowanie bez użycia agenta. Agent Zabbix nie jest potrzebny do kontroli SSH.

Aby wykonywać kontrole SSH, serwer Zabbix musi zostać najpierw skonfigurowany z obsługą SSH2 (libssh lub libssh2). Zobacz także: Wymagania.

> **Attention:**
> Począwszy od RHEL 8 obsługiwany jest tylko libssh. W przypadku innych dystrybucji zalecany jest libssh zamiast libssh2.

Konfiguracja

Uwierzytelnianie za pomocą hasła

Kontrole SSH udostępniają dwie metody uwierzytelniania — parę użytkownik/hasło oraz metodę opartą na pliku klucza.

Jeśli nie zamierzasz używać kluczy, nie jest wymagana żadna dodatkowa konfiguracja, poza podlinkowaniem biblioteki libssh lub libssh2 do Zabbix, jeśli kompilujesz ze źródeł.

Uwierzytelnianie za pomocą pliku klucza

Aby używać uwierzytelniania opartego na kluczu dla pozycji SSH, wymagane są pewne zmiany w konfiguracji serwera.

Otwórz plik konfiguracyjny serwera Zabbix (`zabbix_server.conf`) jako `root` i znajdź następującą linię:

```
##### SSHKeyLocation=
```

Usuń znak komentarza i ustaw pełną ścieżkę do folderu, w którym będą znajdować się klucze publiczny i prywatny:

```
SSHKeyLocation=/home/zabbix/.ssh
```

Zapisz plik, a następnie uruchom ponownie serwer Zabbix.

Ścieżka `/home/zabbix` jest tutaj katalogiem domowym konta użytkownika *zabbix*, a `.ssh` to katalog, w którym domyślnie będą generowane klucze publiczny i prywatny przez polecenie ssh-keygen w katalogu domowym.

Zazwyczaj pakiety instalacyjne serwera Zabbix dla różnych dystrybucji systemu operacyjnego tworzą konto użytkownika *zabbix* z katalogiem domowym w innym miejscu, na przykład `/var/lib/zabbix` (jak w przypadku kont systemowych).

Przed wygenerowaniem kluczy można przenieść katalog domowy do `/home/zabbix`, aby odpowiadał on parametrowi konfiguracji `SSHKeyLocation` serwera Zabbix wspomnianemu powyżej.

Note:
Poniższe kroki można pominąć, jeśli konto *zabbix* zostało dodane ręcznie zgodnie z sekcją instalacji. W takim przypadku katalog domowy konta *zabbix* najprawdopodobniej już znajduje się w `/home/zabbix`.

Aby zmienić katalog domowy konta użytkownika *zabbix*, należy zatrzymać wszystkie procesy robocze, które z niego korzystają:

```
systemctl stop zabbix-agent
systemctl stop zabbix-server
```

Aby zmienić lokalizację katalogu domowego z próbą jego przeniesienia (jeśli istnieje), należy wykonać następujące polecenie:

```
usermod -m -d /home/zabbix zabbix
```

Możliwe jest również, że katalog domowy nie istniał w starej lokalizacji, więc należy go utworzyć w nowym miejscu. Bezpieczna próba wykonania tego wygląda następująco:

```
test -d /home/zabbix || mkdir /home/zabbix
```

Aby upewnić się, że wszystko jest zabezpieczone, można wykonać dodatkowe polecenia ustawiające uprawnienia do katalogu domowego:

```
chown zabbix:zabbix /home/zabbix
chmod 700 /home/zabbix
```

Wcześniej zatrzymane procesy można teraz ponownie uruchomić:

```
systemctl start zabbix-agent
systemctl start zabbix-server
```

Teraz można wykonać kroki generowania kluczy publicznego i prywatnego za pomocą następujących poleceń (dla lepszej czytelności komunikaty wiersza poleceń zostały zakomentowane):

```
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:
Klucze publiczny i prywatny (`id_rsa.pub` i `id_rsa`) zostały domyślnie wygenerowane w katalogu `/home/zabbix/.ssh`, co odpowiada parametrowi konfiguracji `SSHKeyLocation` serwera Zabbix.

Attention:
Typy kluczy inne niż "rsa" mogą być obsługiwane przez narzędzie ssh-keygen i serwery SSH, ale mogą nie być obsługiwane przez libssh2 używany przez Zabbix.

Formularz konfiguracji powłoki

Ten krok należy wykonać tylko raz dla każdego hosta, który będzie monitorowany przez kontrole SSH.

Za pomocą poniższych poleceń można zainstalować plik klucza **publicznego** na zdalnym hoście *10.10.10.10*, aby kontrole SSH mogły być wykonywane przy użyciu konta *root* (dla lepszej czytelności monity poleceń zostały zakomentowane):

```
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.
```

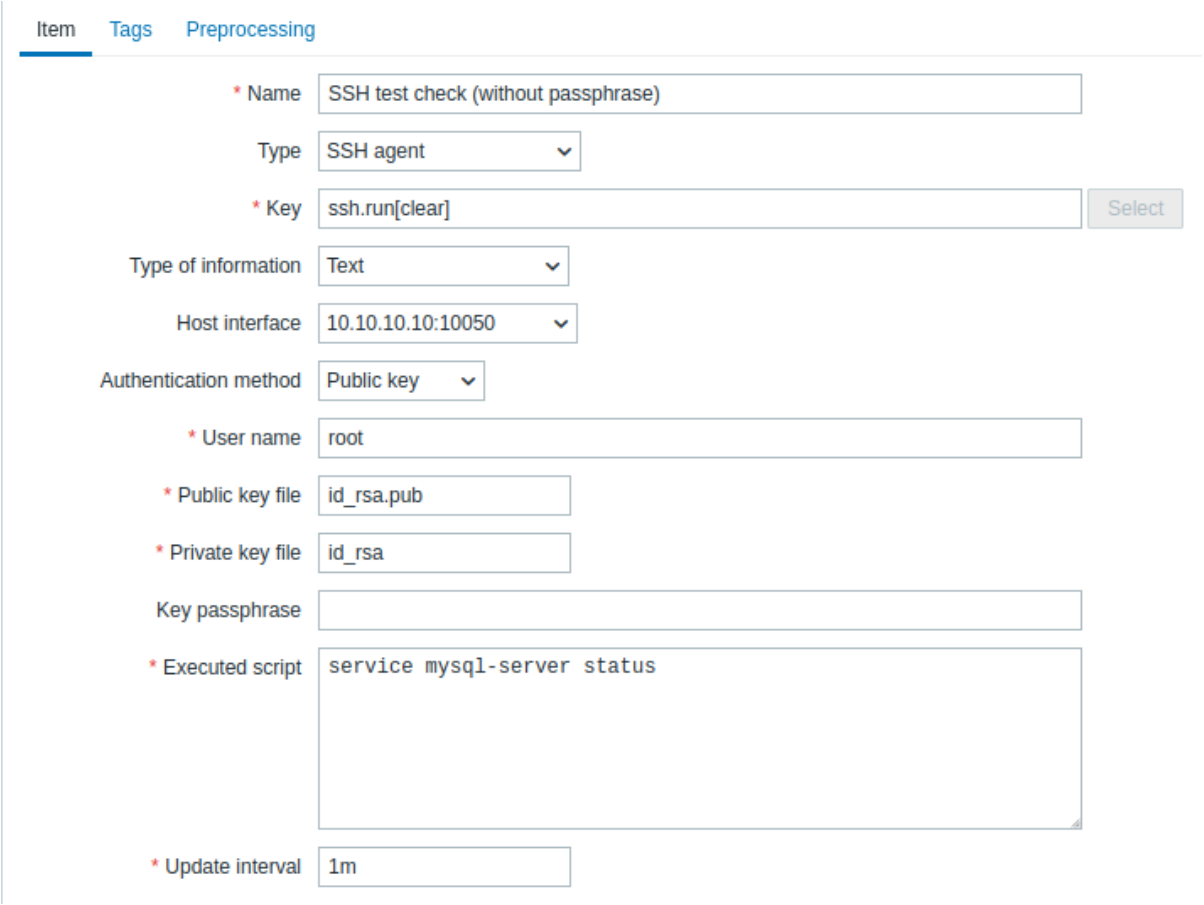
Teraz można sprawdzić logowanie SSH, używając domyślnego klucza prywatnego (`/home/zabbix/.ssh/id_rsa`) dla konta użytkownika *zabbix*:

```
sudo -u zabbix ssh root@10.10.10.10
```

Jeśli logowanie zakończy się powodzeniem, część konfiguracji w powłoce jest zakończona i zdalną sesję SSH można zamknąć.

Konfiguracja pozycji

Rzeczywiste polecenie(-a), które mają zostać wykonane, należy umieścić w polu *Wykonywany skrypt* w konfiguracji pozycji. Wiele poleceń można wykonać jedno po drugim, umieszczając je w nowym wierszu. W takim przypadku zwracane wartości również będą formatowane jako wielowierszowe.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji w przypadku pozycji SSH:

Parametr	Opis	Komentarze
Typ	Wybierz tutaj **SSH agent**.	

Parametr	Opis	Komentarze

Klucz	Unikalny (w obrębie host) klucz pozycji w formacie `ssh.run[unique short description,<ip>,<port>,<encoding>,<ssh options>,<subsystem>]`	**unique short description** jest wymagany i powinien być unikalny dla każdej pozycji SSH w obrębie host.
		
		Domyślny port to 22, a nie port określony w interfejsie, do którego przypisano tę pozycję.
		
		ssh options umożliwiają przekazanie dodatkowych opcji SSH w formacie `key1=value1;key2=value2,value3`. Wiele wartości dla jednego klucza można przekazać, oddzielając je przecinkami (w takim przypadku parametr musi być ujęty w cudzysłów); wiele kluczy opcji można przekazać, oddzielając je średnikami.
		
		Obsługiwane są następujące klucze opcji: `KexAlgorithms`, `HostkeyAlgorithms`, `Ciphers`, `MACs`, `PubkeyAcceptedKeyTypes`. Obsługa klucza opcji i jego wartości zależy od biblioteki SSH (na przykład `PubkeyAcceptedKeyTypes` jest obsługiwany tylko przez libssh); jeśli opcja nie jest obsługiwana, zostanie zwrócony błąd, a pozycja stanie się nieobsługiwana.
		
		Należy pamiętać, że znak „+” służący do dołączania ustawień szyfrów oraz znak „!” służący do wyłączania określonych ustawień szyfrów (tak jak w GnuTLS i OpenSSL) nie są obsługiwane.
		
		Przykłady:
		=>
		`ssh.run[KexAlgorithms,127.0.0.1,,,Ciphers=aes128`
		=>
		`ssh.run[KexAlgorithms,,,,"KexAlgorithms=diffie-h`
		=>
		`ssh.run[PubkeyAcceptedKeyTypes,127.0.0.1,,,Pubke`
		
		subsystem umożliwia przekazanie podsystemu SSH, ograniczając połączenie SSH do określonych operacji dozwolonych przez podsystem (np. transferu plików za pomocą SFTP lub zarządzania urządzeniami sieciowymi za pomocą NETCONF). Należy pamiętać, że użycie podsystemu może również wymagać zastosowania określonej składni skryptu w parametrze *Wykonywany skrypt*.
		
		Przykłady:
		=>
		`ssh.run[SFTPBackup,192.0.2.18,,,,sftp]`
		=>
		`ssh.run[Cisco1234,192.0.2.18,,,,netconf]`

Metoda uwierzytelniania	Jedna z opcji: „Password” lub „Public key”.	
Nazwa użytkownika	Nazwa użytkownika (do 255 znaków) używana do uwierzytelniania na zdalnym hoście. Wymagana.	
Plik klucza publicznego	Nazwa pliku klucza publicznego, jeśli *Metoda uwierzytelniania* to „Public key”. Wymagana.	Przykład: *id_rsa.pub* - domyślna nazwa pliku klucza publicznego wygenerowanego przez polecenie ssh-keygen.
Plik klucza prywatnego	Nazwa pliku klucza prywatnego, jeśli *Metoda uwierzytelniania* to „Public key”. Wymagana.	Przykład: *id_rsa* - domyślna nazwa pliku klucza prywatnego.

Parametr	Opis	Komentarze
Hasło lub *Hasło klucza*	Hasło (do 255 znaków) używane do uwierzytelniania lub hasło klucza **jeśli** zostało użyte dla klucza prywatnego.	Pozostaw pole *Hasło klucza* puste, jeśli hasło klucza nie było używane. Więcej informacji na temat używania haseł kluczy zawiera sekcja znane problemy.

Wykonywany skrypt	Wykonywane polecenie(-a) powłoki za pomocą zdalnej sesji SSH.	Wartość zwracana przez wykonane polecenie(-a) powłoki jest ograniczona do 16 MB (w tym końcowych znaków odstępu, które są obcinane); obowiązują również ograniczenia bazy danych.
		
		Należy pamiętać, że biblioteka libssh2 może obcinać wykonywane skrypty do około 32 kB.
		
		Przykłady:
		`date +%s`
		`systemctl status mysql-server`
		`ps auxww | grep httpd | wc -l`
		
		Przykład (dla podsystemu NETCONF):
		`<rpc>`
		`<get-software-information/>`
		`</rpc>`
		`]]>]]>`
		`<rpc>`
		`<close-session/>`
		`</rpc>`
		`]]>]]>`

Agent Telnet

Przegląd

Kontrole Telnet są wykonywane jako monitorowanie bez agent. Zabbix agent nie jest wymagany do kontroli Telnet.

Konfigurowalne pola

Rzeczywiste polecenie(a) do wykonania należy umieścić w polu **Executed script** w konfiguracji pozycji.
 Wiele poleceń można wykonać jedno po drugim, umieszczając je w nowej linii. W takim przypadku zwracana wartość również będzie sformatowana jako wielowierszowa.

Obsługiwane znaki, którymi może kończyć się znak zachęty powłoki:

- $
- #
- >
- %

Note:
Linia znaku zachęty telnet zakończona jednym z tych znaków zostanie usunięta ze zwracanej wartości, ale tylko dla pierwszego polecenia na liście poleceń, tj. tylko na początku sesji telnet.

Klucz	Opis
telnet.run[<unique short description>,<ip>,<port>,<encoding>]	Uruchamia polecenie na zdalnym urządzeniu przy użyciu połączenia telnet

Attention:
Jeśli sprawdzenie telnet zwraca wartość zawierającą znaki spoza ASCII i w kodowaniu innym niż UTF8, wówczas parametr *<encoding>* w kluczu powinien zostać odpowiednio określony. Więcej szczegółów można znaleźć na stronie encoding of returned values.

agent JMX

Przegląd

Monitorowanie JMX może być używane do monitorowania liczników JMX aplikacji Java.

Monitorowanie JMX ma natywne wsparcie w Zabbix w postaci demona Zabbix o nazwie „Zabbix Java gateway".

Aby pobrać wartość konkretnego licznika JMX na hoście, serwer Zabbix odpytuje **Java gateway** Zabbix, który z kolei używa API zarządzania JMX do zdalnego odpytywania interesującej aplikacji.

Więcej szczegółów oraz informacje o konfiguracji można znaleźć w sekcji Zabbix Java gateway.

Warning:
Komunikacja między Java gateway a monitorowaną aplikacją JMX nie powinna być blokowana przez zaporę sieciową.

Włączanie zdalnego monitorowania JMX dla aplikacji Java

Aplikacja Java nie wymaga instalacji żadnego dodatkowego oprogramowania, ale musi zostać uruchomiona z opcjami wiersza poleceń podanymi poniżej, aby obsługiwać zdalne monitorowanie JMX.

Jako absolutne minimum, jeśli chcesz po prostu rozpocząć monitorowanie prostej aplikacji Java na lokalnym hoście bez wymuszonych zabezpieczeń, uruchom ją z następującymi opcjami:

```
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 /path/to/your/application.jar
```

Spowoduje to, że Java będzie nasłuchiwać przychodzących połączeń JMX na porcie 12345, wyłącznie z lokalnego hosta, oraz nie będzie wymagać uwierzytelniania ani SSL.

Jeśli chcesz zezwolić na połączenia na innym interfejsie, ustaw parametr -Djava.rmi.server.hostname na adres IP tego interfejsu.

Jeśli chcesz bardziej rygorystycznie podejść do kwestii bezpieczeństwa, masz do dyspozycji wiele innych opcji Java. Na przykład kolejny przykład uruchamia aplikację z bardziej wszechstronnym zestawem opcji i udostępnia ją w szerszej sieci, a nie tylko na lokalnym hoście.

```
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 /path/to/your/application.jar
```

Większość (jeśli nie wszystkie) z tych ustawień można określić w pliku `$JRE/lib/management/management.properties` (lub tam, gdzie ten plik znajduje się w twoim systemie).

Zwróć uwagę, że jeśli chcesz używać SSL, musisz zmodyfikować skrypt startup.sh, dodając opcje `-Djavax.net.ssl.*` do Java gateway, aby wiedział, gdzie znaleźć magazyny kluczy i zaufania.

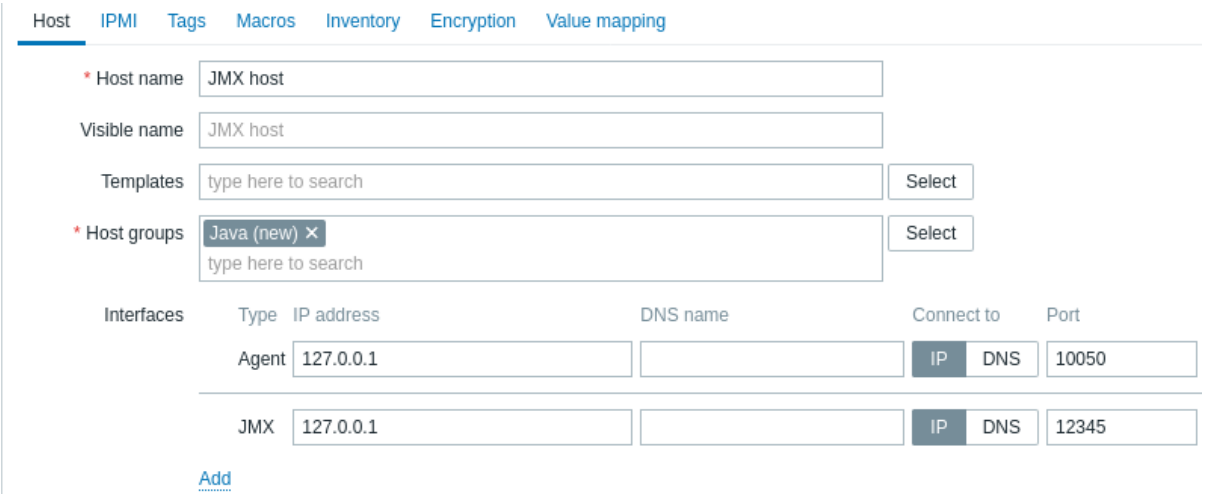
Szczegółowy opis znajduje się w Monitoring and Management Using JMX.

Konfigurowanie interfejsów JMX i pozycji w frontend Zabbix

Gdy Java gateway jest uruchomiony, serwer wie, gdzie go znaleźć, a aplikacja Java została uruchomiona z obsługą zdalnego monitorowania JMX, nadszedł czas, aby skonfigurować interfejsy i pozycje w GUI Zabbix.

Konfigurowanie interfejsu JMX

Zacznij od utworzenia interfejsu typu JMX na interesującym Cię hoście.

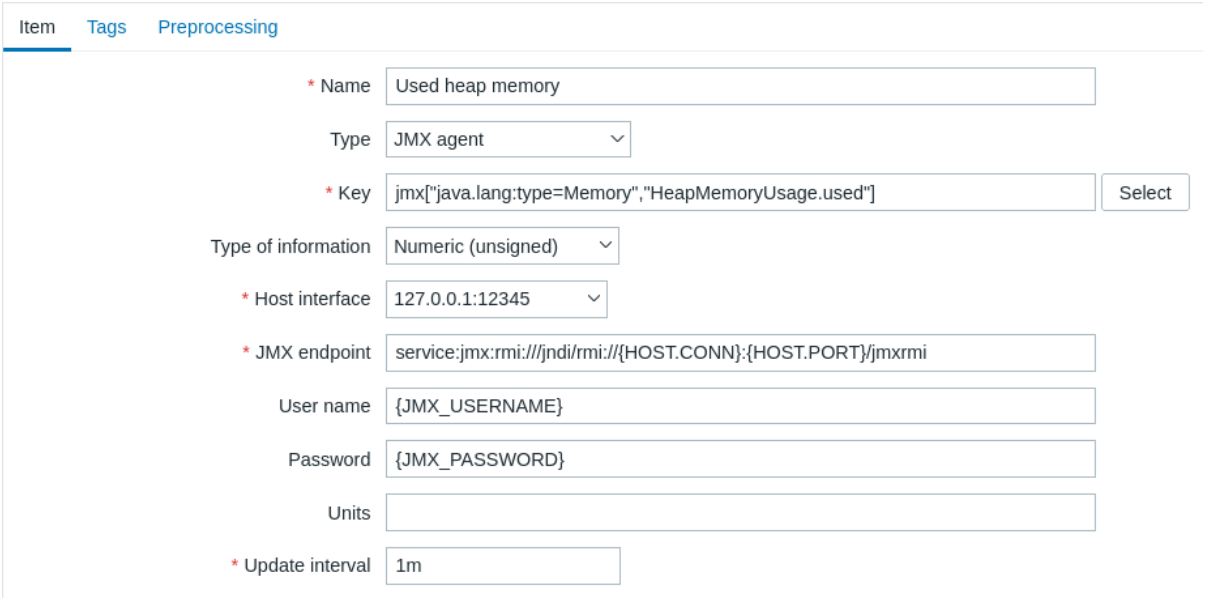


Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Dodawanie pozycji agenta JMX

Dla każdego licznika JMX, który Cię interesuje, dodajesz pozycję **JMX agent** przypisaną do tego interfejsu.

Klucz na poniższym zrzucie ekranu ma postać `jmx["java.lang:type=Memory","HeapMemoryUsage.used"]`.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola, które wymagają podania określonych informacji dla pozycji JMX, to:

Type	Ustaw tutaj **JMX agent**.
Key	Klucz pozycji `jmx[]` zawiera trzy parametry: **object name** - nazwa obiektu MBean **attribute name** - nazwa atrybutu MBean z opcjonalnymi nazwami pól danych złożonych oddzielonymi kropkami **unique short description** - unikalny opis, który umożliwia użycie wielu pozycji JMX z tą samą nazwą obiektu i nazwą atrybutu na hoście (opcjonalnie) Więcej informacji o kluczach pozycji JMX znajduje się poniżej. Możesz wykrywać MBeans i atrybuty MBean za pomocą pozycji `jmx.discovery[]` low-level discovery.
JMX endpoint	Możesz określić niestandardowy punkt końcowy JMX. Upewnij się, że parametry połączenia punktu końcowego JMX są zgodne z interfejsem JMX. Można to osiągnąć, używając makr `{HOST.*}` tak jak w domyślnym punkcie końcowym JMX. Obsługiwane są makra `{HOST.*}` macros oraz makra użytkownika.
User name	Podaj nazwę użytkownika (do 255 znaków), jeśli skonfigurowano uwierzytelnianie w aplikacji Java. Obsługiwane są makra użytkownika.

Password	Podaj hasło (do 255 znaków), jeśli skonfigurowano uwierzytelnianie w aplikacji Java. Obsługiwane są makra użytkownika.

Jeśli chcesz monitorować licznik typu Boolean, który ma wartość ”true” lub ”false”, ustaw typ informacji jako ”Numeric (unsigned)” i wybierz krok preprocessingu ”Boolean to decimal” na karcie Preprocessing. Serwer będzie przechowywać wartości Boolean odpowiednio jako 1 lub 0.

Klucze pozycji JMX bardziej szczegółowo

Proste atrybuty

Nazwa obiektu MBean jest po prostu ciągiem znaków, który definiujesz w swojej aplikacji Java. Z kolei nazwa atrybutu może być bardziej złożona. Jeśli atrybut zwraca prymitywny typ danych (liczbę całkowitą, ciąg znaków itp.), nie ma się czym martwić — klucz będzie wyglądał tak:

```
jmx[com.example:Type=Hello,weight]
```

W tym przykładzie nazwa obiektu to „com.example:Type=Hello”, nazwa atrybutu to „weight”, a typ zwracanej wartości powinien prawdopodobnie być ustawiony na „Numeric (float)”.

Atrybuty zwracające dane złożone

Sytuacja staje się bardziej skomplikowana, gdy atrybut zwraca dane złożone. Na przykład: nazwa atrybutu to ”apple” i zwraca on hash reprezentujący jego parametry, takie jak ”weight”, ”color” itp. Klucz może wyglądać następująco:

```
jmx[com.example:Type=Hello,apple.weight]
```

W ten sposób nazwa atrybutu i klucz hash są rozdzielane przy użyciu symbolu kropki. Analogicznie, jeśli atrybut zwraca zagnieżdżone dane złożone, poszczególne części są rozdzielane kropką:

```
jmx[com.example:Type=Hello,fruits.apple.weight]
```

Atrybuty zwracające dane tabelaryczne

Atrybuty danych tabelarycznych składają się z jednego lub wielu atrybutów złożonych. Jeśli taki atrybut zostanie określony w parametrze nazwy atrybutu, wówczas wartość tej pozycji zwróci kompletną strukturę atrybutu w formacie JSON. Wartości poszczególnych elementów wewnątrz atrybutu danych tabelarycznych można pobrać za pomocą preprocessing.

Przykład atrybutu danych tabelarycznych:

```
jmx[com.example:type=Hello,foodinfo]
```

Wartość pozycji:

```
[
  {
    "a": "apple",
    "b": "banana",
    "c": "cherry"
  },
  {
    "a": "potato",
    "b": "lettuce",
    "c": "onion"
  }
]
```

Problem z kropkami

Jak dotąd wszystko dobrze. Ale co, jeśli nazwa atrybutu lub klucz hash zawiera symbol kropki? Oto przykład:

```
jmx[com.example:Type=Hello,all.fruits.apple.weight]
```

To problem. Jak powiedzieć Zabbixowi, że nazwa atrybutu to „all.fruits”, a nie tylko „all”? Jak odróżnić kropkę, która jest częścią nazwy, od kropki, która oddziela nazwę atrybutu od kluczy hash?

Jest to możliwe, wystarczy poprzedzić kropki będące częścią nazwy ukośnikiem odwrotnym:

```
jmx[com.example:Type=Hello,all\.fruits.apple.weight]
```

W ten sam sposób, jeśli klucz hash zawiera kropkę, należy ją poprzedzić znakiem ucieczki:

```
jmx[com.example:Type=Hello,all\.fruits.apple.total\.weight]
```

Inne problemy

Znak ukośnika odwrotnego w nazwie atrybutu powinien być poprzedzony znakiem ucieczki:

```
jmx[com.example:type=Hello,c:\\documents]
```

Aby obsłużyć wszelkie inne znaki specjalne w kluczu pozycji JMX, zobacz sekcję dotyczącą formatu klucza pozycji section.

To właściwie wszystko. Życzymy udanego monitorowania JMX!

Nieprymitywne typy danych

Możliwa jest praca z niestandardowymi MBeanami zwracającymi nieprymitywne typy danych, które przesłaniają metodę **toString()**.

Korzystanie z niestandardowego endpointu z JBoss EAP 6.4

Niestandardowe endpointy umożliwiają pracę z różnymi protokołami transportowymi innymi niż domyślny RMI.

Aby zilustrować tę możliwość, spróbujmy skonfigurować monitorowanie JBoss EAP 6.4 jako przykład. Najpierw przyjmijmy kilka założeń:

- Masz już zainstalowany Zabbix Java gateway. Jeśli nie, możesz to zrobić zgodnie z dokumentacją.
- serwer Zabbix i Java gateway są zainstalowane z prefiksem `/usr/local/`
- JBoss jest już zainstalowany w `/opt/jboss-eap-6.4/` i działa w trybie standalone
- Załóżmy, że wszystkie te komponenty działają na tym samym hoście
- Firewall i SELinux są wyłączone (lub odpowiednio skonfigurowane)

Wprowadźmy kilka prostych ustawień w `zabbix_server.conf`:

```
JavaGateway=127.0.0.1
StartJavaPollers=5
```

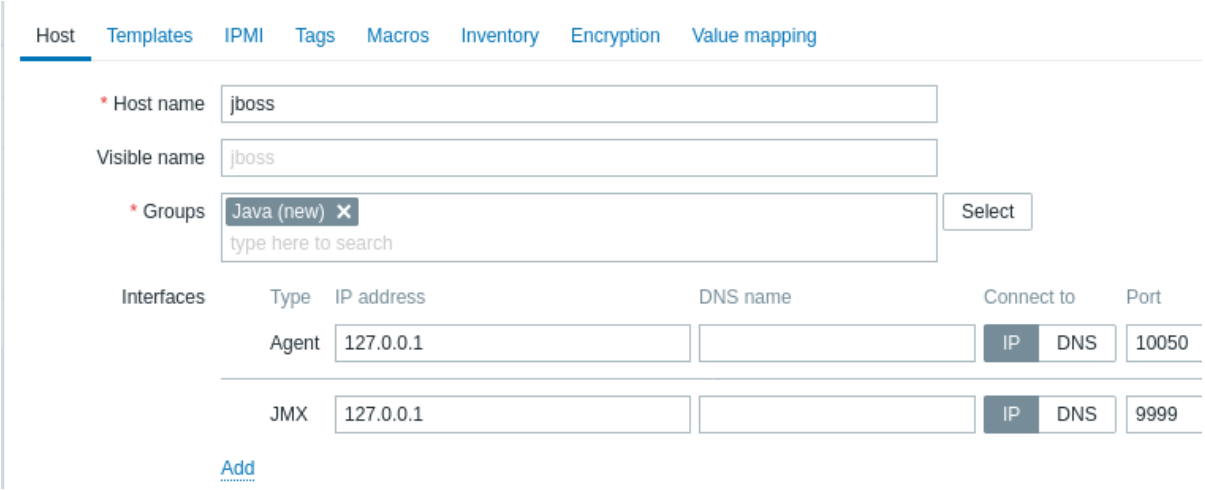
Oraz w pliku konfiguracyjnym `zabbix_java/settings.sh` (lub `zabbix_java_gateway.conf`):

```
START_POLLERS=5
```

Sprawdź, czy JBoss nasłuchuje na swoim standardowym porcie zarządzania:

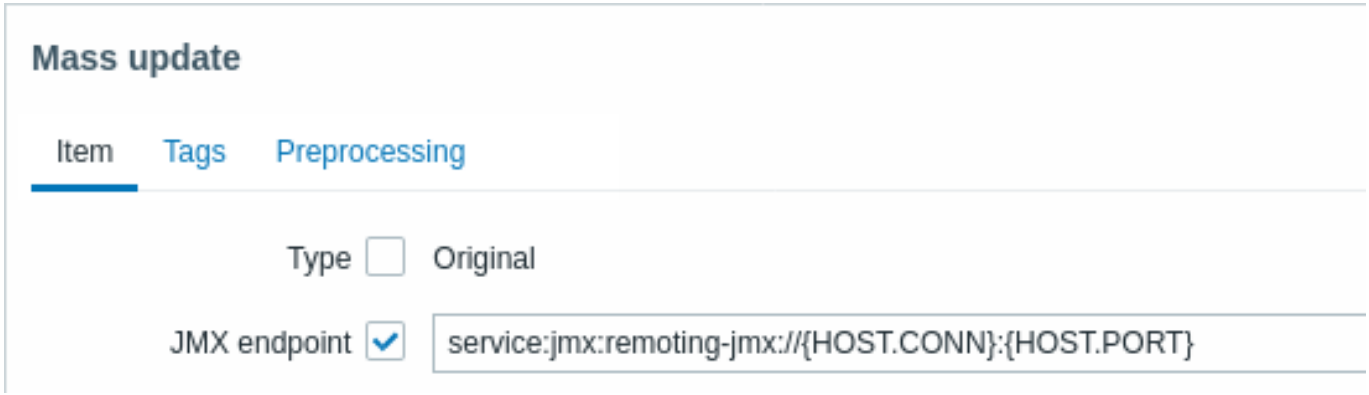
```
$ netstat -natp | grep 9999
tcp        0      0 127.0.0.1:9999          0.0.0.0:*               LISTEN      10148/java
```

Teraz utwórzmy host z interfejsem JMX 127.0.0.1:9999 w Zabbix.



Ponieważ wiemy, że ta wersja JBoss używa protokołu JBoss Remoting zamiast RMI, możemy odpowiednio zbiorczo zaktualizować parametr endpointu JMX dla pozycji w naszym szablonie JMX:

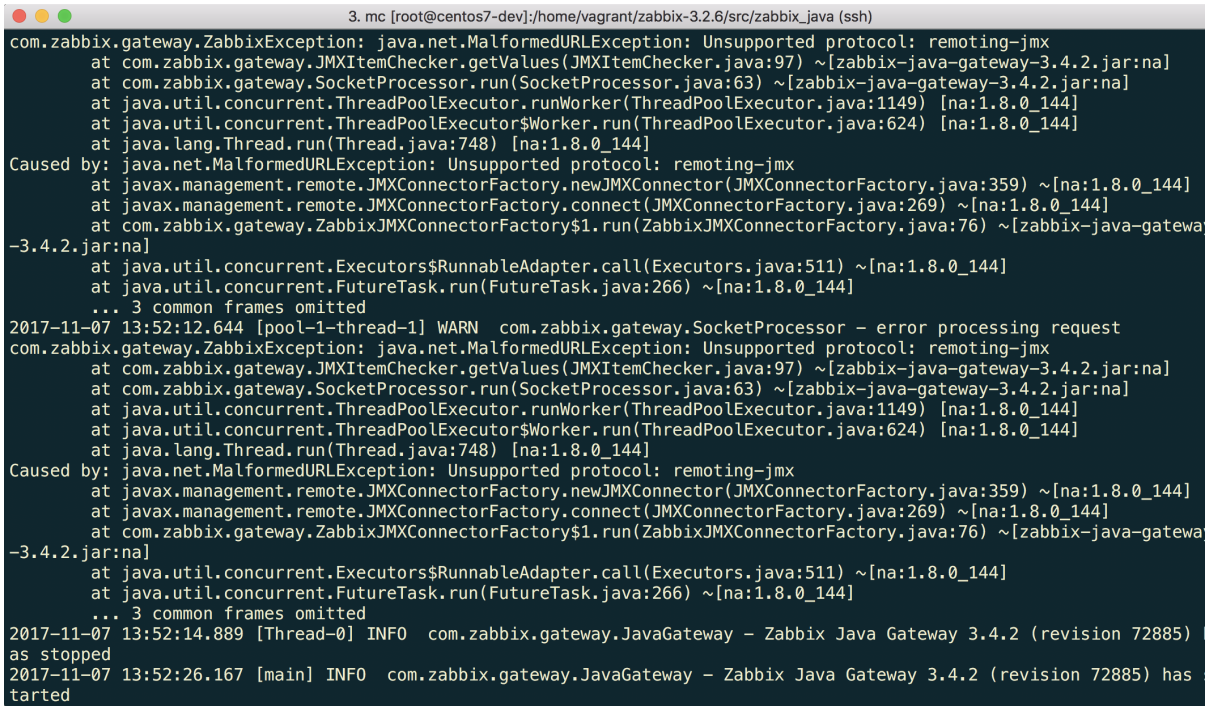
```
service:jmx:remoting-jmx://{HOST.CONN}:{HOST.PORT}
```



Zaktualizujmy pamięć podręczną konfiguracji:

```
/usr/local/sbin/zabbix_server -R config_cache_reload
```

Zwróć uwagę, że najpierw możesz napotkać błąd.



”Unsupported protocol: remoting-jmx” oznacza, że Java gateway nie wie, jak pracować z określonym protokołem. Można to naprawić, tworząc plik `~/needed_modules.txt` o następującej zawartości:

```
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
```

a następnie wykonując polecenie:

```
for i in $(cat ~/needed_modules.txt); do find /opt/jboss-eap-6.4 -iname "${i}*.jar" -exec cp '{}' /usr/loc
```

W ten sposób Java gateway będzie miał wszystkie niezbędne moduły do pracy z jmx-remoting. Pozostaje zrestartować Java gateway, chwilę poczekać i jeśli wszystko zostało wykonane poprawnie, zobaczyć, że dane monitorowania JMX zaczynają napływać do Zabbix (zobacz też: Latest data).

Obliczona pozycja

Przegląd

Ten typ pozycji umożliwia wykonywanie obliczeń na metrykach, które już istnieją. Obliczana pozycja nie zbiera samodzielnie żadnych danych.

Możesz na przykład:

- Zliczać dwie wartości.
- Obliczać średnią z kilku wartości pozycji.
- Obliczać łączną wartość dla grupy pozycji.

Wynikowa wartość jest przechowywana w bazie danych Zabbix tak jak w przypadku każdej innej pozycji; przechowywane są zarówno wartości historii, jak i trendy, a także można generować wykresy.

Obliczenia są oparte na formule. Składnia formuły jest wspólna z wyrażeniami wyzwalaczy.

W rzeczywistości można zachować prostotę wyzwalaczy, umieszczając złożoną logikę przekształcania danych w obliczanej pozycji, a następnie odwołując się do obliczanej pozycji w wyzwalaczu za pomocą klucza pozycji.

Istnieją dwie możliwości określenia źródła danych dla obliczeń:

1. Określenie pozycji w formule.
2. Wykonanie agregacji danych na podstawie filtra wybierającego pozycje.

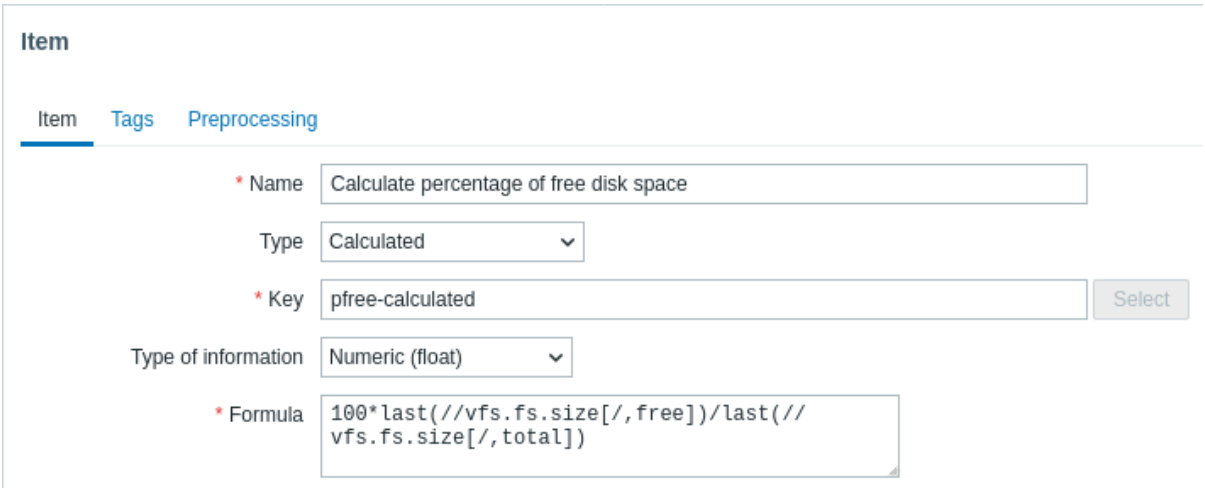
Druga opcja umożliwia obliczanie średnich grupowych, sum itd.

Wszystkie obliczenia są wykonywane przez serwer Zabbix i obsługiwane przez procesy `history poller`; proxy ani agenty nie biorą w nich udziału. Należy pamiętać, że intensywne korzystanie z obliczanych pozycji z długimi okresami może wpływać na wydajność serwera.

Należy pamiętać, że obliczane pozycje mogą uzyskiwać dostęp do historii innych pozycji, w tym pozycji, do których użytkownik tworzący obliczaną pozycję nie ma dostępu. Jeśli takie zachowanie jest niepożądane, można wyłączyć obliczane pozycje, ustawiając parametr konfiguracyjny `StartHistoryPollers` na 0.

Konfigurowalne pola

Aby używać pozycji obliczanych, wybierz typ pozycji **Calculated**.



key to unikatowy identyfikator pozycji (w ramach host). Możesz utworzyć dowolną nazwę klucza, używając obsługiwanych symboli. Klucz umożliwia odwoływanie się do pozycji obliczanej w wyzwalaczach.

Definicję obliczenia należy wprowadzić w polu **Formula**.

Nie ma żadnego związku między formułą a kluczem. Parametry klucza nie są w żaden sposób używane w formule.

W przeciwieństwie do wyrażeń wyzwalaczy, serwer Zabbix przetwarza pozycje obliczane zgodnie z interwałem aktualizacji pozycji, a nie po otrzymaniu nowej wartości.

Pamiętaj, że jeśli jako typ informacji wybierzesz *Numeric (unsigned)*, a wynik obliczenia będzie wartością zmiennoprzecinkową, zostanie on obcięty do liczby całkowitej.

Formuła prostego obliczenia

Składnia prostej formuły jest następująca:

`function(/host/key,<parameter1>,<parameter2>,...)`

gdzie:

Element	Opis
`function`	Jedna z obsługiwanych funkcji: last, min, max, avg, count itd.
`host`	Host pozycji używanej do obliczeń. Bieżący host można pominąć (tj. jak w `function(//key,parameter,...)`).
`key`	Klucz pozycji używanej do obliczeń. Pozycje zwracające wartości z binarnym typem danych lub typem danych JSON nie są obsługiwane. Jeśli zmienisz klucz pozycji pozycji, do której istnieje odwołanie, klucz musi zostać ręcznie zaktualizowany w formule. Wszystkie pozycje, do których odwołują się funkcje historii, muszą istnieć i zbierać dane.
`parameter(s)`	Parametry funkcji, jeśli są wymagane. Obsługiwane są sufiksy czasu oraz sufiksy rozmiaru pamięci.

Bardziej złożona formuła może wykorzystywać kombinację funkcji, operatorów i nawiasów.

Zobacz przykłady formuł pozycji obliczanych:

- (1) Całkowity ruch na interfejsie
- (2) Procent wolnego miejsca na dysku
- (3) Procent ruchu przychodzącego
- (4) Średnia liczba przetworzonych wartości

Niektóre zmienne mogą być przekazywane do formuły za pomocą makr użytkownika. Należy pamiętać, że makra użytkownika:

- będą rozwijane — jeśli są używane do odwołania do parametru funkcji, parametru filtra pozycji lub stałej.
- **nie będą** rozwijane — jeśli odwołują się do funkcji, nazwy hosta, klucza pozycji, parametru klucza pozycji lub operatora.

Dozwolone jest porównywanie ze stringami.

Jeśli w pamięci podręcznej nie ma aktualnych danych i w funkcji nie zdefiniowano okresu odpytywania, Zabbix domyślnie cofnie się maksymalnie o jeden tydzień w przeszłość, aby odpytać bazę danych o wartości historyczne.

Pozycja obliczana może stać się nieobsługiwana w kilku przypadkach:

- pozycja, do której istnieje odwołanie
  - nie została znaleziona
  - jest wyłączona lub należy do wyłączonego hosta
  - nie jest obsługiwana (z wyjątkiem funkcji nodata() oraz operatorów z nieznanymi wartościami)
- brak danych do obliczenia funkcji
- dzielenie przez zero
- użyto nieprawidłowej składni

Agregacje danych

Agregacje danych opierają się na filtrze, który wybiera wiele pozycji. Agregacje danych są również typem pozycji obliczanej, jednak w formule używa się nieco innej składni.

Aby pobrać agregaty, użyj jednej z obsługiwanych funkcji agregujących: `avg`, `max`, `min`, `sum` itd. Następnie dodaj funkcję **foreach** jako jedyny parametr oraz jej filtr pozycji, aby wybrać wymagane pozycje:

```
aggregate_function(function_foreach(/host/key?[group="host group"],timeperiod))
```

Funkcja **foreach** (np. *avg_foreach*, *count_foreach* itd.) zwraca jedną wartość agregowaną dla każdej wybranej pozycji. Pozycje są wybierane przy użyciu filtra pozycji (`/host/key?[group="host group"]`) z historii pozycji. Więcej informacji można znaleźć w sekcji funkcje foreach.

Jeśli niektóre pozycje nie mają danych dla żądanego okresu, są ignorowane podczas obliczeń. Jeśli żadna pozycja nie ma danych, funkcja zwróci błąd.

Alternatywnie można podać kilka pozycji jako parametry agregacji:

```
aggregate_function(function(/host/key,parameter),function(/host2/key2,parameter),...)
```

Zwróć uwagę, że `function` musi tutaj być funkcją historii/trendu.

Zobacz przykłady obliczeń agregujących:

- (5) Całkowity ruch przychodzący dla hosta
- (6) Całkowita przestrzeń dyskowa dla grupy
- (7) Średnie obciążenie procesora dla grupy
- (8) Średnie obciążenie procesora dla grupy i tagu
- (9) Średnia liczba przetworzonych wartości
- (10) Średnia liczba zapytań dla grupy
- (11) Całkowita liczba pomyślnych kontroli DNS
- (12) Całkowita liczba nieobsługiwanych pozycji dla grupy
- (13) Statystyki interfejsu dla grupy

Makra użytkownika i makra wykrywania niskopoziomowego są obsługiwane w:

- parametrach klucza pozycji
- parametrach funkcji
- warunkach filtra (nazwa grupy hostów i nazwa tagu)
- stałych wyrażenia

Obliczenie agregujące może stać się nieobsługiwane, jeśli:

- nie zostanie znaleziona żadna z odwoływanych pozycji (co może się zdarzyć, jeśli klucz pozycji jest nieprawidłowy, żadna z pozycji nie istnieje lub wszystkie uwzględnione grupy są nieprawidłowe)
- brak danych do obliczenia funkcji

Przykłady pozycji obliczanych

(1) Całkowity ruch na interfejsie

```
last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes])
```

Oblicza całkowitą przepustowość na eth0.

(2) Procent wolnego miejsca na dysku

```
100*last(//vfs.fs.size[/,free])/last(//vfs.fs.size[/,total])
```

Zabbix pobierze najnowsze wartości wolnego i całkowitego miejsca na dysku (na `/`) i obliczy wartość procentową zgodnie z podanym wzorem.

(3) Procent ruchu przychodzącego

```
100*last(//net.if.in[eth0,bytes])/(last(//net.if.in[eth0,bytes])+last(//net.if.out[eth0,bytes]))
```

Oblicz procent ruchu przychodzącego w całkowitym ruchu.

(4) Średnia liczba przetworzonych wartości

```
avg(/Zabbix server/zabbix[wcache,values],10m)
```

Oblicza 10-minutową średnią liczby wartości przetworzonych przez Zabbix.

(5) Całkowity ruch przychodzący dla hosta

```
sum(last_foreach(/host/net.if.in[*]))
```

Oblicza sumę wszystkich pozycji pasujących do `net.if.in[*]` na hoście.

(6) Całkowita przestrzeń dyskowa dla grupy

```
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"]))
```

Oblicz całkowitą przestrzeń dyskową dla wszystkich `MySQL Servers`.

(7) Średnie obciążenie procesora dla grupy

```
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"]))
```

Oblicz średnie obciążenie procesora dla wszystkich `MySQL Servers`.

(8) Średnie obciążenie procesora dla grupy i tagu

```
avg(last_foreach(/*/system.cpu.load?[(group="Servers A" or group="Servers B" or group="Servers C") and (ta
```

Oblicz średnie obciążenie CPU na wszystkich hostach w wielu grupach hostów, które mają określone tagi.

(9) Średnia liczba zapytań dla grupy

```
avg(avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m))
```

Oblicz 5-minutową średnią liczby zapytań na sekundę dla `MySQL Servers`.

(10) Łączna liczba udanych sprawdzeń DNS

```
sum(last_foreach(/*/net.dns[*,*,*]))
```

Oblicz łączną liczbę udanych sprawdzeń DNS we wszystkich hostach. Pokazana forma pozycji to `net.dns[192.0.2.1,example.com,A]` jako przykład jednego z możliwych kluczy.

Zwróć uwagę, że symbole wieloznaczne muszą odpowiadać liczbie parametrów w kluczu (tutaj `net.dns` ma trzy parametry: ip, name, type).

(11) Łączna liczba nieobsługiwanych pozycji dla grupy

```
sum(last_foreach(/*/zabbix[host,,items_unsupported]?[group="Zabbix servers"]))
```

Oblicz łączną liczbę nieobsługiwanych pozycji dla `Zabbix servers`.

(12) Statystyki interfejsu dla grupy

```
sum(last_foreach(/*/net.if.out[eth0,bytes]?[group="video"])) / sum(last_foreach(/*/nginx_stat.sh[active]?[
```

Oblicz statystyki interfejsu dla grupy `video`.

Przykłady poprawnej/niepoprawnej składni

Wyrażenia (w tym wywołania funkcji) nie mogą być używane jako parametry historii, trendów ani funkcji foreach function. Jednak same te funkcje mogą być używane w parametrach innych (niehistorycznych) funkcji.

Wyrażenie	Przykład
Poprawne	`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)))`
Niepoprawne	`sum(/host/key,10+2)`
	`sum(/host/key, avg(10,2))`
	`sum(/host/key,last(/host/key2))`

Należy pamiętać, że w wyrażeniu takim jak:

```
sum(sum_foreach(//resptime[*],5m))/sum(count_foreach(//resptime[*],5m))
```

(gdzie `resptime[*]` jest przykładem wzorca klucza pozycji pasującego do kluczy `resptime[1]`, `resptime[2]` itd., używanego do uzyskiwania czasu odpowiedzi z kilku aktywnych węzłów klastra) nie można zagwarantować, że obie części równania będą zawsze zawierać ten sam zestaw wartości. Podczas obliczania jednej części wyrażenia może nadejść nowa wartość dla żądanego okresu, a wtedy druga część wyrażenia będzie zawierać inny zestaw wartości.

Pozycja zależna

Przegląd

Zdarzają się sytuacje, gdy jedna pozycja zbiera jednocześnie wiele metryk lub gdy bardziej sensowne jest równoczesne zbieranie powiązanych metryk, na przykład:

- wykorzystanie CPU poszczególnych rdzeni
- przychodzący/wychodzący/całkowity ruch sieciowy

Aby umożliwić zbiorcze pobieranie metryk i ich jednoczesne użycie w kilku powiązanych pozycjach, Zabbix obsługuje pozycje zależne. Pozycje zależne są zależne od pozycji nadrzędnej, która zbiera ich dane jednocześnie, w ramach jednego zapytania. Nowa wartość pozycji nadrzędnej automatycznie uzupełnia wartości pozycji zależnych. Pozycje zależne nie mogą mieć innego interwału aktualizacji niż pozycja nadrzędna.

Opcje preprocessing w Zabbix mogą być używane do wyodrębnienia części potrzebnej dla pozycji zależnej z danych pozycji nadrzędnej.

Preprocessing jest zarządzany przez proces `preprocessing manager` wraz z wątkami roboczymi wykonującymi kroki preprocessing. Wszystkie wartości z preprocessing, otrzymane od różnych mechanizmów zbierania danych, przechodzą przez preprocessing manager przed dodaniem do pamięci podręcznej historii. Między mechanizmami zbierania danych (pollerami, trapperami itp.) a procesem preprocessing wykorzystywana jest komunikacja IPC oparta na gniazdach.

Kroki preprocessing oraz przetwarzanie pozycji zależnych są wykonywane przez Zabbix serwer lub Zabbix proxy (jeśli host jest monitorowany przez proxy).

Pozycje dowolnego typu, nawet pozycje zależne, mogą być ustawione jako pozycje nadrzędne. Dodatkowe poziomy pozycji zależnych mogą być używane do wyodrębniania mniejszych części z wartości istniejącej pozycji zależnej.

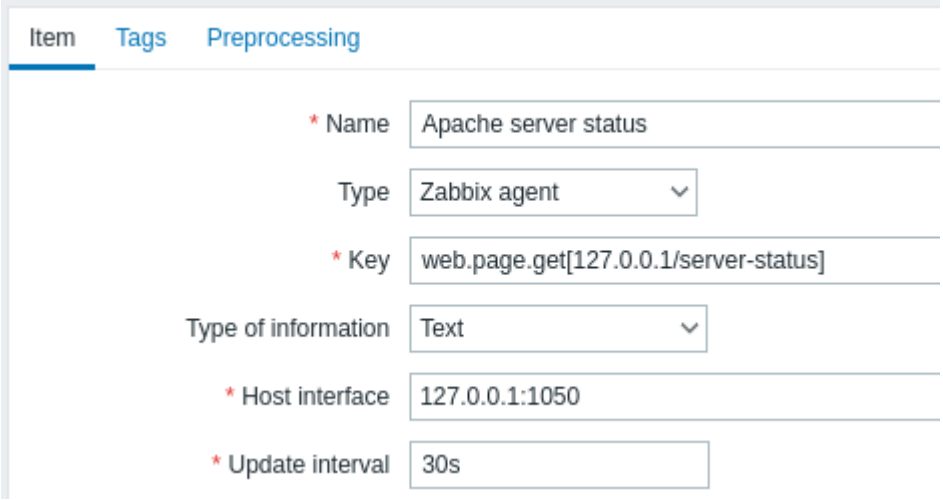
Ograniczenia

- Dozwolone są tylko zależności w obrębie tego samego hosta (szablonu)
- Prototyp pozycji może zależeć od innego prototypu pozycji lub zwykłej pozycji z tego samego hosta
- Zależna pozycja na hoście z główną pozycją z szablonu nie zostanie wyeksportowana do XML

Konfiguracja pozycji

Pozycja zależna zależy od swojej pozycji nadrzędnej w zakresie danych. Dlatego najpierw musi zostać skonfigurowana (lub już istnieć) **pozycja nadrzędna**:

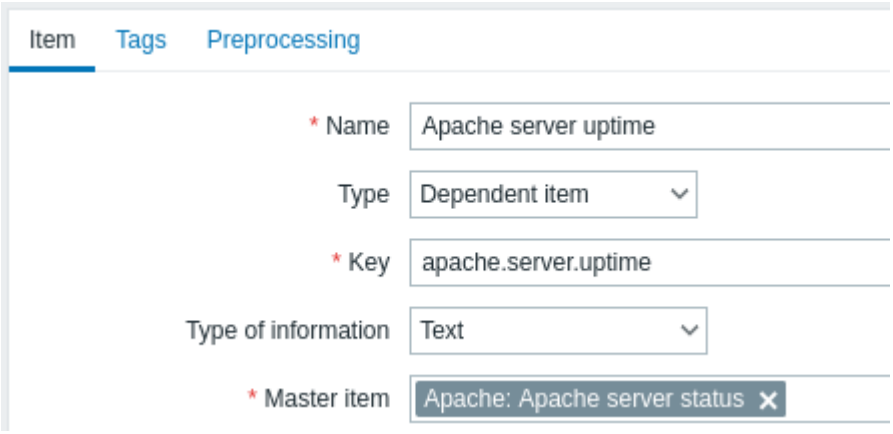
1. Przejdź do *Zbieranie danych* > *Hosty*.
2. Kliknij *Pozycje* w wierszu hosta.
3. Kliknij *Utwórz pozycję*.
4. Wprowadź parametry pozycji w formularzu.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Kliknij *Dodaj*, aby zapisać pozycję nadrzędną.

Następnie możesz skonfigurować **pozycję zależną**.



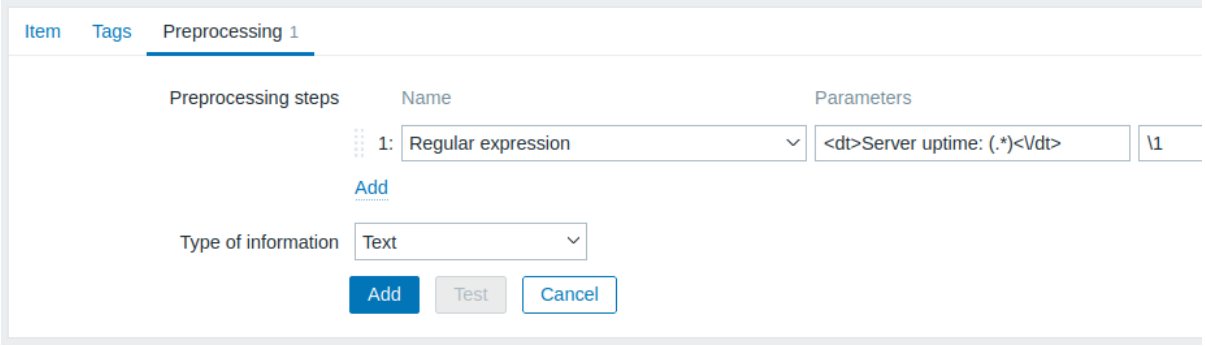
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające podania szczególnych informacji dla pozycji zależnych to:

Type	Wybierz tutaj **Dependent item**.

Key	Wprowadź klucz, który będzie używany do identyfikacji pozycji.
Master item	Wybierz pozycję nadrzędną. Wartość pozycji nadrzędnej zostanie użyta do wypełnienia wartości pozycji zależnej.
Type of information	Wybierz typ informacji, który będzie odpowiadał formatowi danych przechowywanych w tej pozycji.

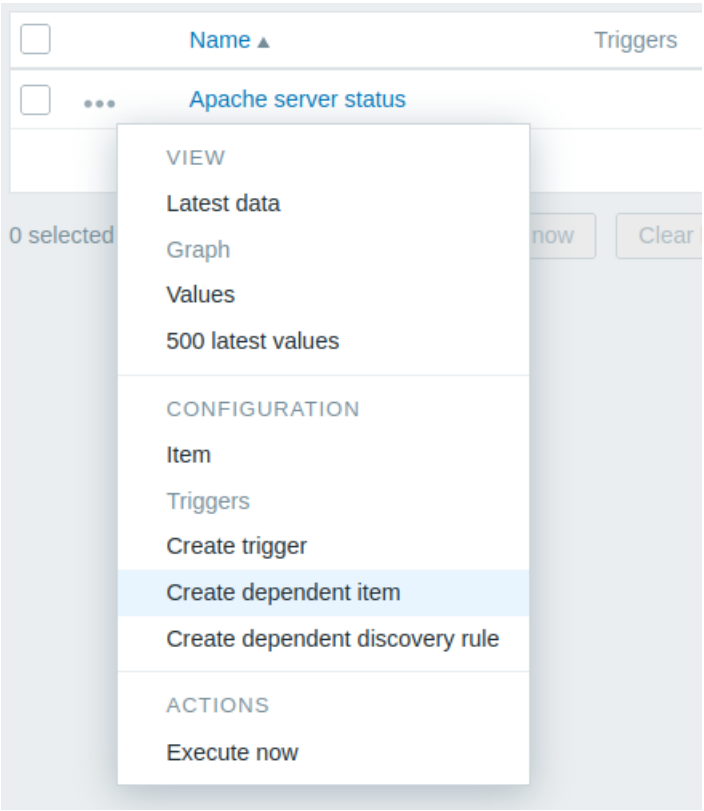
Możesz użyć przetwarzania wstępnego wartości pozycji preprocessing, aby wyodrębnić wymaganą część wartości pozycji nadrzędnej.



Bez przetwarzania wstępnego wartość pozycji zależnej będzie dokładnie taka sama jak wartość pozycji nadrzędnej.

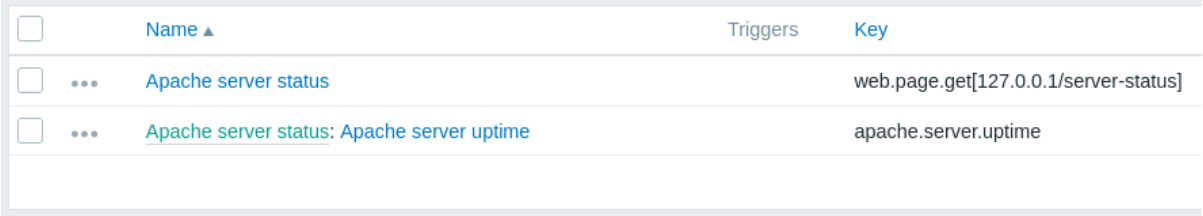
Kliknij *Dodaj*, aby zapisać pozycję zależną.

Skrót do szybszego tworzenia pozycji zależnej jest dostępny po kliknięciu przycisku ••• na liście pozycji i wybraniu *Utwórz pozycję zależną*.



Wyświetlanie

Na liście pozycji pozycje zależne są wyświetlane z nazwą nadrzędnej pozycji jako prefiksem.



Jeśli nadrzędna pozycja zostanie usunięta, wszystkie jej pozycje zależne również zostaną usunięte.

Pozycja skryptu

Przegląd

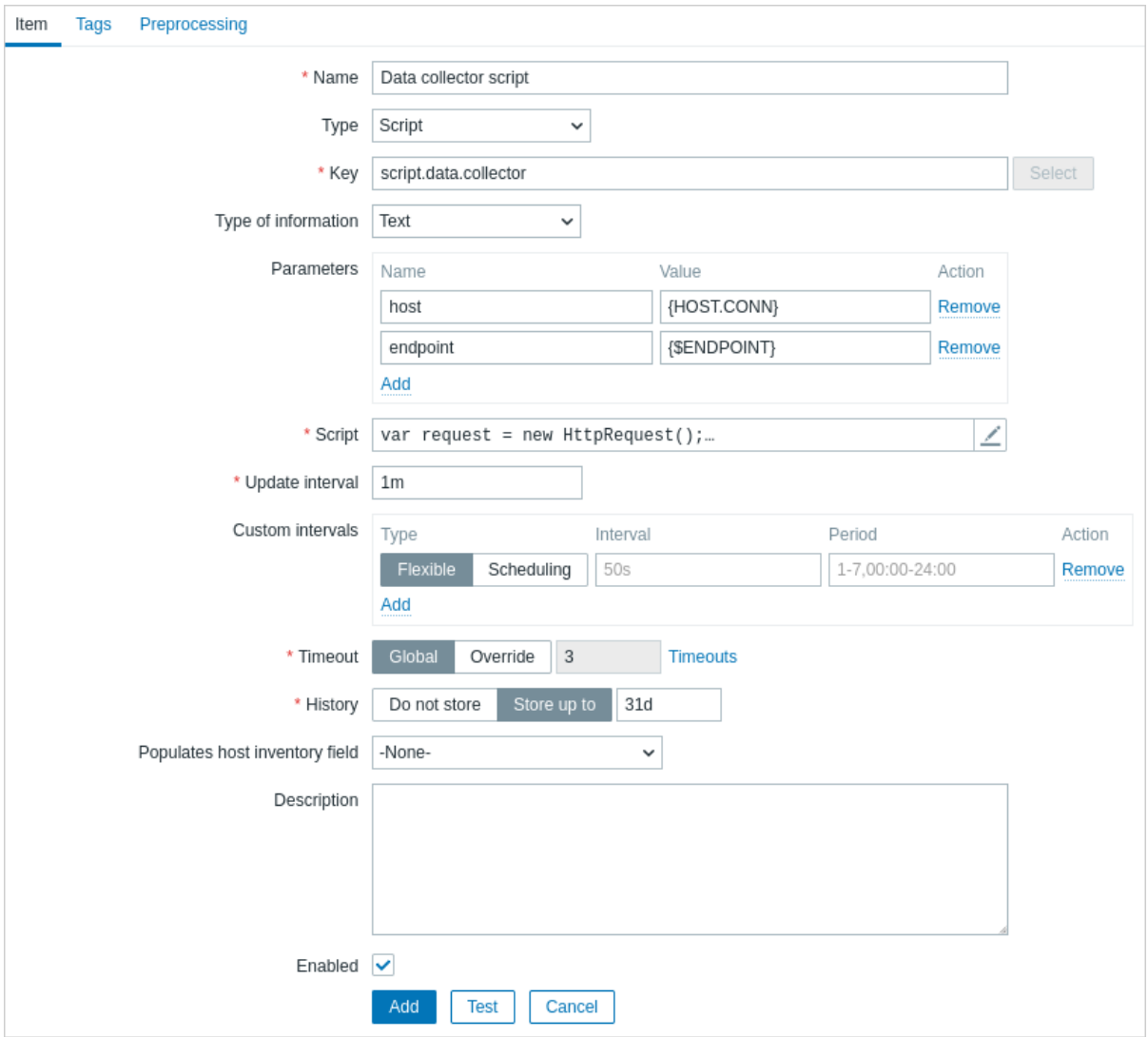
Pozycje skryptowe mogą być używane do zbierania danych przez wykonywanie zdefiniowanego przez użytkownika kodu JavaScript z możliwością pobierania danych przez HTTP/HTTPS. Oprócz skryptu można określić opcjonalną listę parametrów (par nazw i wartości) oraz limit czasu.

Ten typ pozycji może być przydatny w scenariuszach zbierania danych, które wymagają wielu kroków lub złożonej logiki. Na przykład pozycja skryptowa może być skonfigurowana tak, aby wykonać wywołanie HTTP, następnie w pewien sposób przetworzyć dane otrzymane w pierwszym kroku i przekazać przekształconą wartość do drugiego wywołania HTTP.

Pozycje skryptowe są przetwarzane przez pollery serwera lub proxy Zabbix.

Konfiguracja

W polu *Typ* formularza konfiguracji pozycji wybierz Script, a następnie wypełnij wymagane pola.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji typu Script to:

Pole	Opis
Key	Wprowadź unikalny klucz, który będzie używany do identyfikacji pozycji.
Parameters	Określ zmienne przekazywane do skryptu jako pary atrybut-wartość. Obsługiwane są makra użytkownika. Aby sprawdzić, które wbudowane makra są obsługiwane, wyszukaj „Script-type item” w tabeli obsługiwanych makr.

Pole	Opis
Script	Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w pole parametru lub ikonę ołówka obok niego. Kod ten musi zawierać logikę zwracania wartości metryki. Kod ma dostęp do wszystkich parametrów oraz wszystkich dodatkowych obiektów JavaScript dodanych przez Zabbix. Zobacz także: Przewodnik po JavaScript.
Timeout	Limit czasu wykonywania JavaScript (1-600 s; jego przekroczenie spowoduje zwrócenie błędu). Pamiętaj, że w zależności od skryptu wyzwolenie limitu czasu może potrwać dłużej. Więcej informacji o parametrze *Timeout* znajdziesz w sekcji ogólnych atrybutów pozycji.

Przykłady

Proste zbieranie danych

Pobierz zawartość *https://www.example.com/release_notes*:

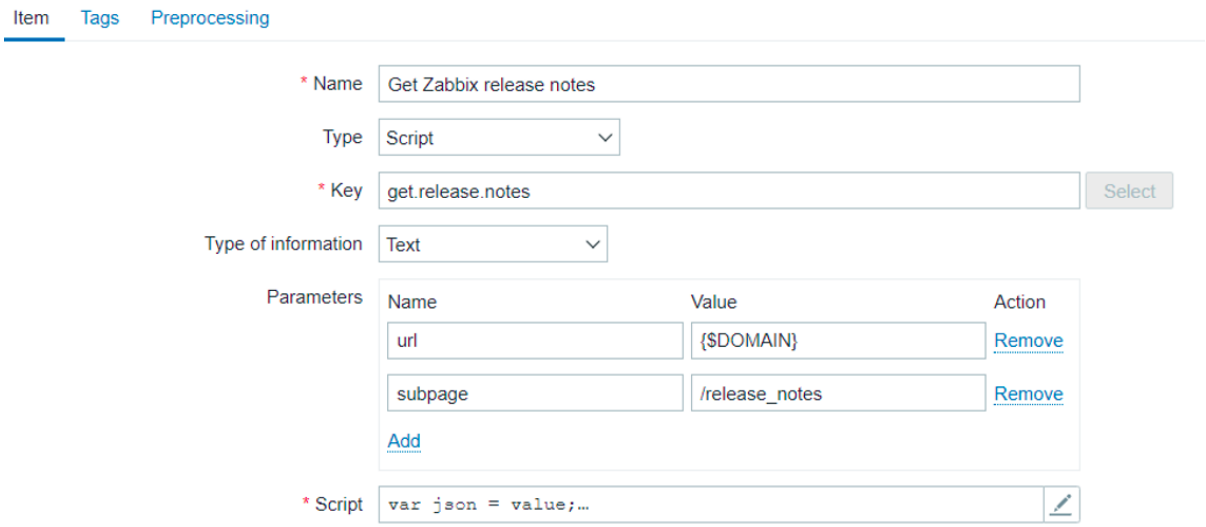
- Utwórz pozycję typu „Script".
- W polu *Script* wprowadź:

```
var request = new HttpRequest();
return request.get("https://www.example.com/release_notes");
```

Zbieranie danych z użyciem parametrów

Pobierz zawartość określonej strony i wykorzystaj parametry:

- Utwórz pozycję typu "Script" z dwoma parametrami:
  - **url : {$DOMAIN}** (makro użytkownika {$DOMAIN} powinno być zdefiniowane, najlepiej na poziomie hosta)
  - **subpage : /release_notes**



- W polu *Script* wprowadź:

```
var obj = JSON.parse(value);
var url = obj.url;
var subpage = obj.subpage;
var request = new HttpRequest();
return request.get(url + subpage);
```

Wiele żądań HTTP

Pobierz zawartość zarówno *https://www.example.com*, jak i *https://www.example.com/release_notes*:

- Utwórz pozycję typu „Script".
- W polu *Script* wprowadź:

```
var request = new HttpRequest();
return request.get("https://www.example.com") + request.get("https://www.example.com/release_notes");
```

Logowanie

Dodaj wpis „Log test" do logu serwera Zabbix i w odpowiedzi odbierz wartość pozycji „1":

- Utwórz pozycję typu „Script”.
- W polu *Script* wprowadź:

```
Zabbix.log(3, 'Log test');
return 1;
```

Pozycja przeglądarki

Przegląd

Pozycje przeglądarki umożliwiają monitorowanie złożonych witryn internetowych i aplikacji webowych za pomocą przeglądarki.

> **Attention:**
> Obsługa pozycji przeglądarki jest obecnie eksperymentalna.

Pozycje przeglądarki zbierają dane, wykonując zdefiniowany przez użytkownika kod JavaScript i pobierając dane przez HTTP/HTTPS. Ta pozycja może symulować takie działania związane z przeglądarką, jak klikanie, wprowadzanie tekstu, nawigowanie po stronach internetowych oraz inne interakcje użytkownika z witrynami internetowymi lub aplikacjami webowymi.

Oprócz skryptu można określić opcjonalną listę parametrów (pary nazwy i wartości) oraz limit czasu.

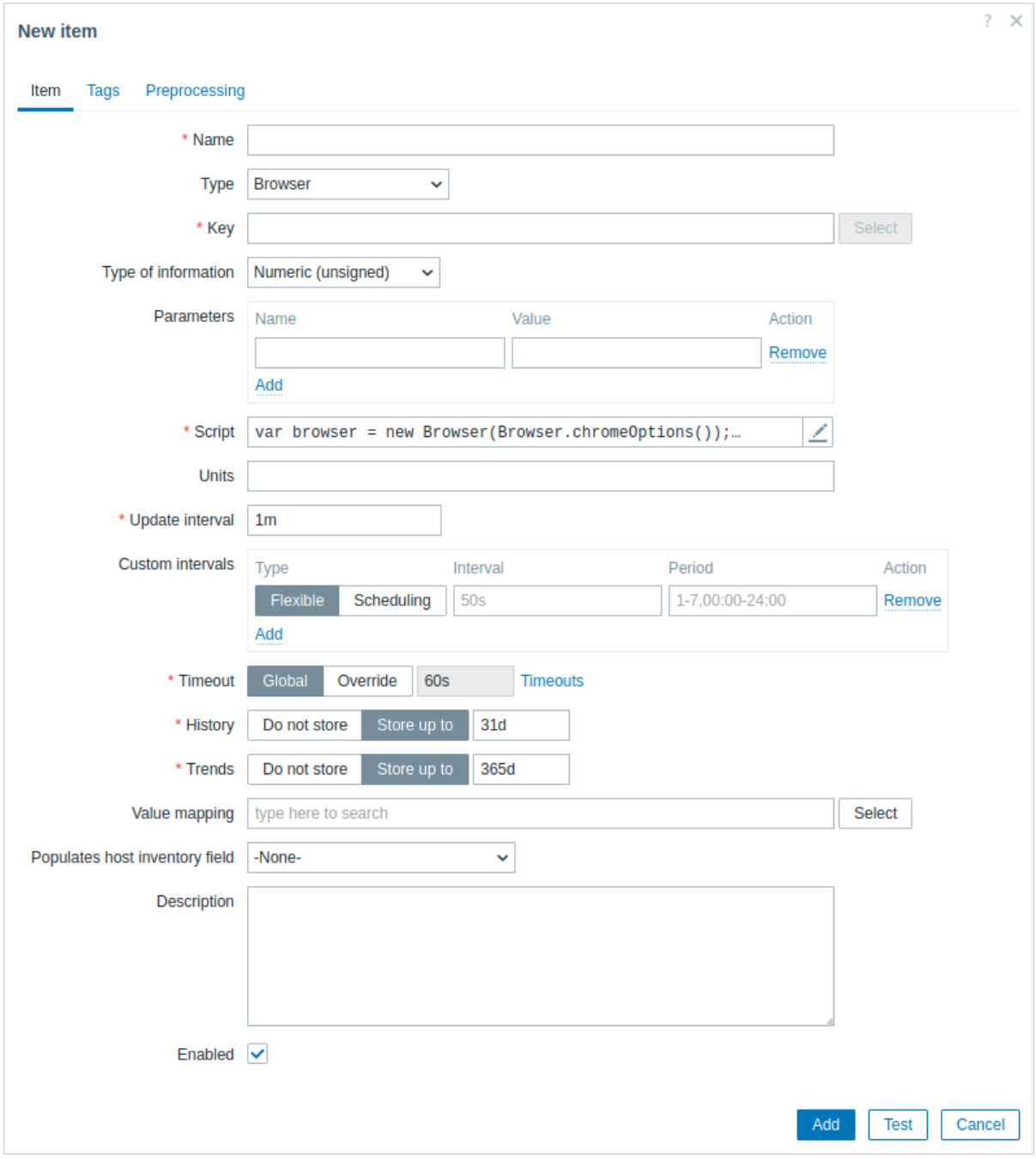
> **Attention:**
> Ta pozycja częściowo implementuje standard W3C WebDriver z użyciem Selenium Server lub zwykłego WebDrivera (na przykład ChromeDriver) jako punktu końcowego testów webowych. Aby pozycja działała, ustaw parametr `WebDriverURL` w pliku konfiguracyjnym Zabbix serwer/proxy (jeśli używasz ChromeDriver, zobacz Security Considerations). Aby uzyskać lepszą wydajność, rozważ użycie dedykowanego serwera dla środowiska testów webowych.

Sprawdzenia pozycji przeglądarki są wykonywane i przetwarzane przez pollery przeglądarki serwera Zabbix lub proxy. W razie potrzeby można dostosować liczbę wstępnie utworzonych instancji pollerów przeglądarki w parametrze `StartBrowserPollers` pliku konfiguracyjnego Zabbix serwer/proxy.

Do monitorowania złożonych witryn internetowych i aplikacji webowych dostępny jest szablon Website by Browser jako szablon gotowy do użycia.

Konfiguracja

W polu *Type* formularza konfiguracji pozycji wybierz Browser, a następnie wypełnij wymagane pola.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Pola wymagające określonych informacji dla pozycji Browser to:

Field	Description
Key	Wprowadź unikalny klucz, który będzie używany do identyfikacji pozycji.
Parameters	Określ zmienne, które mają zostać przekazane do skryptu jako pary atrybut-wartość. Obsługiwane są makra użytkownika. Aby sprawdzić, które wbudowane makra są obsługiwane, wyszukaj frazę ”Browser-type item” w tabeli obsługiwanych makr.
Script	Wprowadź kod JavaScript w modalnym edytorze, który otwiera się po kliknięciu w pole parametru lub ikonę ołówka obok niego. Kod ten musi zawierać logikę zwracania wartości metryki. Kod ma dostęp do wszystkich parametrów, wszystkich dodatkowych obiektów JavaScript oraz obiektów JavaScript pozycji Browser dodanych przez Zabbix. Zobacz także: Przewodnik po JavaScript.
Timeout	Limit czasu wykonania JavaScript (1-600 s; jego przekroczenie spowoduje zwrócenie błędu). Należy pamiętać, że w zależności od skryptu wyzwolenie limitu czasu może potrwać dłużej. Więcej informacji o parametrze *Timeout* można znaleźć w sekcji ogólnych atrybutów pozycji.

Przykłady

Przykład konfiguracji Zabbixa do monitorowania witryn internetowych przy użyciu szablonu *Website by Browser* można znaleźć w artykule Monitor websites with Browser items.

Domyślny skrypt

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Przechodzi pod określony adres URL.
3. Zbiera wpisy wydajności i statystyki sesji, a następnie zwraca je jako ciąg JSON.

W polu *Skrypt* wprowadź:

```
var browser = new Browser(Browser.chromeOptions());

try {
    browser.navigate("http://example.com");
    browser.collectPerfEntries();
}
finally {
    return JSON.stringify(browser.getResult());
}
```

Inicjalizacja przeglądarki z niestandardowymi capabilities

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki dla dostępnej przeglądarki na podstawie pierwszej pasującej przeglądarki zgodnie z kolejnością określoną w skrypcie.
2. Definiuje capabilities przeglądarki, w tym strategię ładowania strony oraz opcje specyficzne dla każdej przeglądarki, takie jak tryb headless dla przeglądarek Chrome, Firefox i Microsoft Edge.

W polu *Script* wprowadź:

```
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"
            }
```

```
        ]
    }
});
```

Inicjalizacja przeglądarki z GUI

Domyślnie sesje przeglądarki (z wyjątkiem Safari) są inicjalizowane w trybie headless, co oznacza, że graficzny interfejs użytkownika (GUI) przeglądarki nie jest wyświetlany.

Poniższy skrypt inicjalizuje sesję przeglądarki z włączonym GUI.

Pamiętaj, że jeśli WebDriver nie może zlokalizować pliku binarnego przeglądarki, możesz ręcznie określić ścieżkę.

```
var opts = Browser.chromeOptions();
opts.capabilities.alwaysMatch['goog:chromeOptions'].args = [];

// Aby zainicjalizować sesję Firefox z GUI, odkomentuj poniższe linie:
// var opts = Browser.firefoxOptions();
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].binary = 'usr/bin/firefox';
// opts.capabilities.alwaysMatch['moz:firefoxOptions'].args = [];

// Aby zainicjalizować sesję Microsoft Edge z GUI, odkomentuj poniższe linie:
// 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:
Jeśli testy są uruchamiane na zdalnym serwerze lub w kontenerze, możesz użyć klienta Virtual Network Computing (VNC), aby połączyć się z serwerem VNC tej maszyny. Pozwala to zdalnie wyświetlać GUI przeglądarki i wchodzić z nim w interakcję.

Wykonywanie zrzutów ekranu

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Ustawia rozmiar obszaru roboczego przeglądarki w celu określenia rozmiaru zrzutu ekranu (określonego jako parametry, patrz poniżej).
3. Przechodzi do adresu URL (określonego jako parametr, patrz poniżej).
4. Zbiera statystyki sesji, wykonuje zrzut ekranu i dodaje go do zebranych statystyk.
5. Obsługuje błędy, przechwytując komunikaty o błędach i zrzut ekranu.
6. Zwraca zebrane wyniki jako ciąg JSON.

Skrypt używa również parametrów z formularza konfiguracji pozycji:

- webURL - http://example.com
- width - 1920
- height - 1080

W polu *Script* wprowadź:

```
var browser, result;

var browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); // Parse the JSON string containing parameters passed from Zabbix.

    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);
}
```

Sprawdzenie logowania do Zabbix

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Przechodzi do strony (określonej jako parametr, patrz poniżej).
3. Wprowadza nazwę użytkownika i hasło (określone jako parametry, patrz poniżej).
4. Wyszukuje i klika przycisk logowania.
5. Wyszukuje i klika przycisk wylogowania.
6. Zbiera dane wydajności przed logowaniem i po nim, a także po wylogowaniu.
7. Obsługuje błędy przez przechwytywanie komunikatów o błędach i zrzutu ekranu.
8. Zwraca zebrane wyniki jako ciąg JSON.

Skrypt używa również parametrów z formularza konfiguracji pozycji:

- webURL - http://{HOST.CONN}/index.php
- username - {$USERNAME}
- password - {$PASSWORD}

W polu *Script* wprowadź:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value); // Parse the JSON string containing parameters passed from Zabbix.

    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);
}
```

Znajdowanie linków

Poniższy skrypt:

1. Inicjalizuje sesję przeglądarki.
2. Definiuje funkcję usuwania zduplikowanych elementów z tablicy (zobacz krok 5).
3. Przechodzi do strony (określonej przez parametry, patrz poniżej).
4. Znajduje linki na stronie.
5. Usuwa zduplikowane linki, aby zapewnić ich unikalność.
6. Wyodrębnia tylko linki zaczynające się od ”http”.
7. Formatuje wyodrębnione linki do określonej struktury.
8. Obsługuje błędy przez przechwytywanie komunikatów o błędach i zrzutu ekranu.
9. Zwraca zebrane wyniki jako ciąg JSON.

Skrypt używa również parametrów z formularza konfiguracji pozycji:

- scheme - {$WEBSITE.SCHEME}
- domain - {$WEBSITE.DOMAIN}
- path - {$WEBSITE.PATH}

W polu *Script* wprowadź:

```
var browser, result;

browser = new Browser(Browser.chromeOptions());

try {
    var params = JSON.parse(value);  // Parse the JSON string containing parameters passed from Zabbix.

    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);
}
```

Zapytanie telemetryczne

Ten typ pozycji umożliwia pobieranie lub agregowanie danych telemetrycznych. Dane telemetryczne są używane do monitorowania wydajności aplikacji (APM), co pomaga sprawdzać przepływ żądań przez aplikacje i usługi, mierzyć ich wydajność oraz znajdować przyczyny powolnych lub nieudanych operacji.

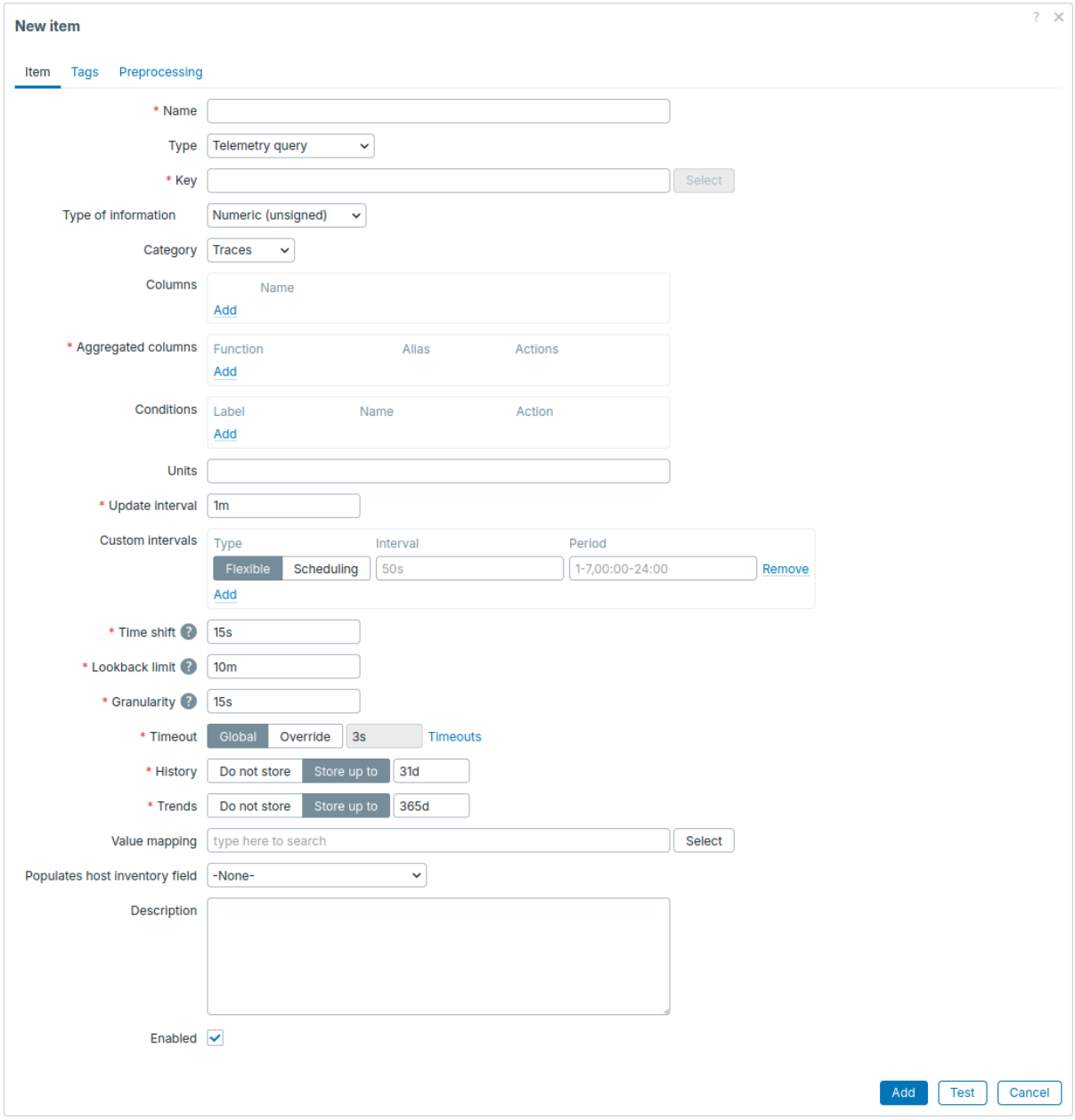
Dane telemetryczne występują w trzech typach: ślady (ścieżka, którą żądanie pokonuje w aplikacjach i usługach), metryki (pomiary, takie jak czas odpowiedzi) oraz dzienniki (tekstowe zapisy zdarzeń). Za pomocą pozycji zapytania telemetrycznego można na przykład znaleźć wszystkie ślady, w których usługa zwróciła błąd, uzyskać średni czas odpowiedzi na podstawie metryk lub wyszukać w dziennikach określony komunikat o błędzie.

Podczas konfigurowania pozycji zapytania telemetrycznego definiuje się zestaw instrukcji (zapytanie), które informują Zabbix, o jakie dane telemetryczne należy poprosić, jak je filtrować i jak je agregować. Po wykonaniu sprawdzenia pozycji Zabbix wysyła to zapytanie do źródła danych telemetrycznych i pobiera wynik na jego podstawie.

Aby użyć tej pozycji, określ źródło danych telemetrycznych i ustaw parametr `StartTelemetryQueryPollers` w konfiguracji serwera Zabbix lub proxy Zabbix, zależnie od tego, który z nich monitoruje host pozycji.

Konfiguracja

Podczas tworzenia pozycji ustaw jej typ na *Telemetry query*:



Parametr	Opis
Nazwa	Zobacz parametry wspólne dla wszystkich pozycji.
Typ	Wybierz *Telemetry query*.
Klucz	Zobacz parametry wspólne dla wszystkich pozycji.
Typ informacji	
Kategoria	Wybierz typ sygnału, który chcesz odpytywać: *Traces*, *Metrics* lub *Logs*.
Punkty metryki	Wybierz typ punktu metryki, który chcesz odpytywać: *Sum*, *Gauge*, *Histogram* lub *Exponential histogram*. Ten parametr jest dostępny, jeśli dla opcji *Kategoria* ustawiono wartość *Metrics*.
Kolumny	Dodaj kolumny danych telemetrycznych, które chcesz odpytywać. W przypadku kolumn zawierających pary klucz-wartość wprowadź również nazwę klucza. Rekordy z tych kolumn zostaną uwzględnione w wartości pozycji bez zmian.

Parametr	Opis
Kolumny zagregowane	Dodaj kolumny danych telemetrycznych, które chcesz odpytywać, oraz określ sposób obliczania wartości na podstawie ich rekordów: **Funkcja** - wybierz funkcję stosowaną do rekordów kolumny: *min*, *max*, *avg*, *count*, *sum* lub *percentile*. **Procent** - wprowadź percentyl obliczany na podstawie rekordów kolumny (jeśli dla opcji *Funkcja* ustawiono wartość *percentile*). **Kolumna** - wybierz kolumnę, na podstawie której ma zostać wykonane obliczenie (jeśli dla opcji *Funkcja* nie ustawiono wartości *count*). **Alias** - wprowadź unikatową nazwę obliczonej wartości, używaną do identyfikowania jej w wartości pozycji.
Typ obliczenia	Wybierz sposób oceny warunków ustawionych w poniższym parametrze *Warunki*: **And/Or** - wszystkie warunki muszą być spełnione; warunki o tym samym *Typie* i *Kluczu atrybutu* są łączone operatorem *Or*. **And** - wszystkie warunki muszą być spełnione. **Or** - co najmniej jeden warunek musi być spełniony. **Wyrażenie niestandardowe** - warunki muszą być spełnione zgodnie z formułą, na przykład `A and (B or C)`. Ten parametr jest dostępny, jeśli ustawiono co najmniej dwa warunki.
Warunki	Dodaj warunki filtrowania rekordów kolumn danych telemetrycznych: **Typ** - wybierz kolumnę danych telemetrycznych, do której będzie stosowany warunek. **Klucz atrybutu** - wprowadź klucz, do którego będzie stosowany warunek (jeśli dla opcji *Typ* ustawiono kolumnę zawierającą pary klucz-wartość). **Operator** - wybierz porównanie, które ma być używane (zobacz poniżej) między wartością rekordu a *Wartością atrybutu*. **Wartość atrybutu** - wprowadź wartość, z którą ma być porównywana wartość rekordu (jeśli dla opcji *Operator* nie ustawiono wartości *Exists*). Jeśli dla opcji *Typ* ustawiono kolumnę z pojedynczą wartością, warunek będzie stosowany do rekordu kolumny; jeśli kolumna zawiera tablicę wartości, warunek zostanie spełniony dla rekordu, gdy co najmniej jedna wartość w tablicy spełnia ten warunek. Jeśli dla opcji *Typ* ustawiono kolumnę zawierającą pary klucz-wartość, warunek będzie stosowany do wartości klucza określonego w polu *Klucz atrybutu*; jeśli kolumna zawiera tablicę par klucz-wartość, warunek zostanie spełniony dla rekordu, gdy co najmniej jedna para z określonym *Kluczem atrybutu* spełnia ten warunek. W przypadku kolumn z pojedynczą wartością można używać następujących operatorów: **Equals** - dopasowuje rekordy równe *Wartości atrybutu* (z uwzględnieniem wielkości liter). **Does not equal** - dopasowuje rekordy różne od *Wartości atrybutu* (z uwzględnieniem wielkości liter). **Contains** - dopasowuje rekordy zawierające *Wartość atrybutu* (z uwzględnieniem wielkości liter). **Does not contain** - dopasowuje rekordy niezawierające *Wartości atrybutu* (z uwzględnieniem wielkości liter). W przypadku kolumn zawierających pary klucz-wartość można używać następujących operatorów: **Equals** - dopasowuje rekordy, w których istnieje *Klucz atrybutu*, a jego wartość jest równa *Wartości atrybutu* (z uwzględnieniem wielkości liter). **Does not equal** - dopasowuje rekordy, w których istnieje *Klucz atrybutu*, a jego wartość różni się od *Wartości atrybutu* (z uwzględnieniem wielkości liter). **Exists** - dopasowuje rekordy, w których istnieje *Klucz atrybutu* (z uwzględnieniem wielkości liter).
Jednostki *Interwał aktualizacji* *Interwały niestandardowe*	Zobacz parametry wspólne dla wszystkich pozycji.

Parametr	Opis
Przesunięcie czasu	Wprowadź, jak daleko w przeszłość (przed bieżący czas) ma sięgać koniec zakresu czasu, dla którego pozycja wykonuje zapytanie. Obsługiwana jest liczba sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub makra użytkownika. Każde zapytanie pobiera dane z określonego zakresu czasu. Zakres zwykle rozpoczyna się w miejscu, w którym zakończył się zakres czasu poprzedniego zapytania, i kończy się w punkcie określonym przez opcję *Przesunięcie czasu*. Jeśli zakres czasu poprzedniego zapytania zakończył się później niż bieżący czas pomniejszony o *Granularność*, zakres czasu bieżącego zapytania rozpoczyna się w bieżącym czasie pomniejszonym o *Granularność*, przez co nakłada się na zakres czasu poprzedniego zapytania. Jeśli nie ma poprzedniego zapytania, od którego można kontynuować, zakres rozpoczyna się w punkcie określonym przez opcję *Limit wstecznego wyszukiwania*. Jeśli host tej pozycji zacznie być monitorowany przez inną jednostkę (np. inny proxy z grupy proxy), pozycja może nie wiedzieć, kiedy zakończyło się poprzednie zapytanie. Może to powodować duplikowanie wartości.
Limit wstecznego wyszukiwania	Wprowadź, jak daleko w przeszłość (przed bieżący czas) pozycja ma sięgać po dane, gdy nie ma poprzedniego zapytania, od którego można kontynuować (np. podczas pierwszego uruchomienia pozycji), lub gdy poprzednie zapytanie jest starsze niż *Interwał aktualizacji* (np. ponieważ serwer Zabbix był niedostępny). Obsługiwana jest liczba sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub makra użytkownika. Jeśli źródło danych telemetrycznych zawiera dużą ilość danych (np. ponad milion rekordów na godzinę), duża wartość *Limitu wstecznego wyszukiwania* może znacznie obniżyć wydajność podczas korzystania z niej przez pozycję.
Granularność	Wprowadź długość fragmentu, na który dzielony jest pełny zakres czasu zapytania. Obsługiwana jest liczba sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub makra użytkownika.
Limit czasu *Historia* *Trendy* *Mapowanie wartości* *Wypełnia pole inwentaryzacji hosta* *Opis* *Włączone*	Zobacz parametry wspólne dla wszystkich pozycji.

Obsługiwane kolumny danych telemetrycznych

W tej sekcji wymieniono kolumny obsługiwane w następujących parametrach pozycji: *Columns*, *Aggregated columns* i *Conditions*. Są to kolumny używane przez ClickHouse do przechowywania danych telemetrycznych zgodnie ze schematem eksportera ClickHouse OpenTelemetry.

Wybierz *Category*:

- Ślady
- Metryki: suma
- Metryki: wskaźnik
- Metryki: histogram
- Metryki: histogram wykładniczy
- Dzienniki

Ślady

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
Czas trwania	✓	✓	–
Events.Attributes	–	–	✓
Events.Name	–	–	✓
ParentSpanId	✓	–	✓
ResourceAttributes	✓	–	✓
ScopeName	✓	–	✓
ScopeVersion	✓	–	✓

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
ServiceName	✓	–	✓
SpanAttributes	✓	–	✓
SpanId	✓	–	✓
SpanKind	✓	–	✓
SpanName	✓	–	✓
StatusCode	✓	–	✓
StatusMessage	✓	–	✓
Znacznik czasu	✓	✓	–
TraceId	✓	–	✓
TraceState	✓	–	✓

Metryki: suma

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
AggregationTemporality	✓	–	–
Attributes	✓	–	✓
Exemplars.FilteredAttributes	–	–	✓
Flags	✓	–	–
IsMonotonic	✓	–	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
TimeUnix	✓	✓	–
Value	✓	✓	–

Metryki: Gauge

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
Atrybuty	✓	–	✓
Exemplars.FilteredAttributes	–	–	✓
Flagi	✓	–	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
TimeUnix	✓	✓	–
Value	✓	✓	–

Metryki: Histogram

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
AggregationTemporality	✓	–	–
Attributes	✓	–	✓
Count	✓	✓	–
Exemplars.FilteredAttributes	–	–	✓
Flags	✓	–	–
Max	✓	✓	–
MetricDescription	✓	–	✓
MetricName	✓	–	✓
MetricUnit	✓	–	✓
Min	✓	✓	–
ResourceAttributes	✓	–	✓
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeDroppedAttrCount	✓	✓	–
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
ServiceName	✓	–	✓
StartTimeUnix	✓	✓	–
Sum	✓	✓	–
TimeUnix	✓	✓	–

Metryki: histogram wykładniczy

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
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	✓	✓	–

Dzienniki

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
Body	✓	–	✓
EventName	✓	–	✓
LogAttributes	✓	–	✓
ResourceAttributes	✓	–	✓

Kolumna	*Kolumny*	*Kolumny zagregowane*	*Warunki*
ResourceSchemaUrl	✓	–	✓
ScopeAttributes	✓	–	✓
ScopeName	✓	–	✓
ScopeSchemaUrl	✓	–	✓
ScopeVersion	✓	–	✓
SpanId	✓	–	✓
ServiceName	✓	–	✓
SeverityNumber	✓	✓	–
SeverityText	✓	–	✓
Timestamp	✓	✓	–
TraceFlags	✓	–	–
TraceId	✓	–	✓

Historia i trendy

Przegląd

Historia i trendy to dwa sposoby przechowywania zebranych danych w Zabbix.

Podczas gdy historia przechowuje każdą zebraną wartość, trendy przechowują uśrednione informacje w ujęciu godzinowym i dlatego zużywają mniej zasobów.

Przechowywanie historii

Możesz ustawić, przez ile dni historia będzie przechowywana:

- we formularzu właściwości pozycji
- podczas masowej aktualizacji pozycji
- podczas konfigurowania zadań housekeepera

Wszelkie starsze dane zostaną usunięte przez housekeeper.

Ogólna, zdecydowana rekomendacja jest taka, aby przechowywać historię przez możliwie najmniejszą liczbę dni i w ten sposób nie przeciążać bazy danych dużą liczbą wartości historycznych.

Zamiast przechowywać długą historię, możesz dłużej przechowywać dane trendów. Na przykład możesz przechowywać historię przez 14 dni, a trendy przez 5 lat.

Możesz uzyskać dobre wyobrażenie o tym, ile miejsca wymagają dane historii w porównaniu z danymi trendów, odwołując się do strony dotyczącej rozmiaru bazy danych.

Przy krótszym okresie przechowywania historii nadal będzie można przeglądać starsze dane na wykresach, ponieważ do wyświetlania starszych danych wykresy będą używać wartości trendów.

Attention:
Jeśli historia jest ustawiona na '0', pozycja będzie aktualizować tylko pozycje zależne i inwentarz. Żadne funkcje wyzwalaczy nie będą obliczane, ponieważ obliczanie wyzwalaczy opiera się wyłącznie na danych historii.

Note:
Jako alternatywny sposób zachowania historii rozważ użycie funkcji eksportu historii modułów ładowalnych.

Przechowywanie trendów

Trendy to wbudowany mechanizm redukcji danych historycznych, który przechowuje minimalną, maksymalną, średnią oraz całkowitą liczbę wartości dla każdej godziny dla numerycznych typów danych.

Możesz ustawić, przez ile dni trendy będą przechowywane:

- we formularzu właściwości pozycji
- podczas masowej aktualizacji pozycji
- podczas konfigurowania zadań Housekeepera

Trendy zwykle mogą być przechowywane znacznie dłużej niż historia. Wszelkie starsze dane zostaną usunięte przez housekeeper.

Serwer Zabbix gromadzi dane trendów w czasie działania w pamięci podręcznej trendów, w miarę napływu danych. Serwer zapisuje trendy z **poprzedniej godziny** każdej pozycji do bazy danych (gdzie frontend może je znaleźć) w następujących sytuacjach:

- serwer otrzymuje pierwszą wartość pozycji z bieżącej godziny
- do końca bieżącej godziny pozostało 5 minut lub mniej, a nadal nie ma żadnych wartości pozycji z bieżącej godziny
- serwer zatrzymuje się

Aby zobaczyć trendy na wykresie, należy poczekać co najmniej do początku następnej godziny (jeśli pozycja jest aktualizowana często) i co najwyżej do końca następnej godziny (jeśli pozycja jest aktualizowana rzadko), czyli maksymalnie 2 godziny.

Gdy serwer zapisuje pamięć podręczną trendów i w bazie danych istnieją już trendy dla tej godziny (na przykład serwer został uruchomiony ponownie w połowie godziny), serwer musi używać instrukcji aktualizacji zamiast prostych instrukcji wstawiania. Dlatego w większych instalacjach, jeśli wymagane jest ponowne uruchomienie, zaleca się zatrzymanie serwera pod koniec jednej godziny i uruchomienie go na początku następnej godziny, aby uniknąć nakładania się danych trendów.

Tabele historii w żaden sposób nie uczestniczą w generowaniu trendów.

Attention:
Jeśli dla trendów ustawiono wartość „0”, serwer Zabbix w ogóle nie oblicza ani nie przechowuje trendów.

Note:
Trendy są obliczane i przechowywane z użyciem tego samego typu danych co oryginalne wartości. W rezultacie obliczenia wartości średniej dla wartości typu unsigned są zaokrąglane, a im mniejszy jest przedział wartości, tym mniej precyzyjny będzie wynik. Na przykład, jeśli pozycja ma wartości 0 i 1, wartość średnia wyniesie 0, a nie 0,5.
Ponadto ponowne uruchomienie serwera może spowodować utratę precyzji obliczeń wartości średniej dla typu unsigned w bieżącej godzinie.

Parametry użytkownika

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.

Wynik polecenia

Wartością zwracaną przez polecenie jest standardowe wyjście wraz ze standardowym wyjściem błędów wygenerowanym przez polecenie.

Attention:
Pozycja zwracająca tekst (informacje typu znakowego, log lub tekstowego) nie stanie się nieobsługiwana w przypadku standardowego wyjścia błędów.

Wartość zwracana jest ograniczona do 16 MB (włącznie z końcowymi białymi znakami, które są obcinane); obowiązują również ograniczenia bazy danych.

Parametry użytkownika zwracające tekst (informacje typu znakowego, log lub tekstowego) mogą również zwracać białe znaki. W przypadku nieprawidłowego wyniku pozycja stanie się nieobsługiwana.

Przykłady prostych parametrów użytkownika

Proste polecenie:

```
UserParameter=ping,echo 1
```

agent będzie zawsze zwracał '1' dla pozycji z kluczem 'ping'.

Bardziej złożony przykład:

```
UserParameter=mysql.ping,mysqladmin -uroot ping | grep -c alive
```

agent zwróci '1', jeśli serwer MySQL działa, w przeciwnym razie '0'.

Przykład 1

Coś bardzo prostego:

```
UserParameter=ping[*],echo $1
```

Możemy zdefiniować nieograniczoną liczbę pozycji do monitorowania, wszystkie w formacie ping[coś].

- ping[0] - zawsze zwróci '0'
- ping[aaa] - zawsze zwróci 'aaa'

Przykład 2

Dodajmy więcej sensu!

```
UserParameter=mysql.ping[*],mysqladmin -u$1 -p$2 ping | grep -c alive
```

Ten parametr może być używany do monitorowania dostępności bazy danych MySQL. Możemy przekazać nazwę użytkownika i hasło:

```
mysql.ping[zabbix,our_password]
```

Przykład 3

Ile linii w pliku pasuje do wyrażenia regularnego?

```
UserParameter=wc[*],grep -c "$2" $1
```

Ten parametr może być używany do obliczania liczby linii w pliku.

```
wc[/etc/passwd,root]
wc[/etc/services,zabbix]
```

Rozszerzanie agentów Zabbix

Ten samouczek zawiera instrukcje krok po kroku dotyczące rozszerzania funkcjonalności agenta Zabbix za pomocą user parameter.

Krok 1

Napisz skrypt lub polecenie w wierszu poleceń, aby pobrać wymagany parametr.

Na przykład możemy napisać następujące polecenie, aby uzyskać łączną liczbę zapytań wykonanych przez serwer MySQL:

```
mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

Po wykonaniu polecenie zwraca łączną liczbę zapytań SQL.

Krok 2

Dodaj polecenie do pliku konfiguracyjnego agenta Zabbix (`zabbix_agentd.conf`):

```
UserParameter=mysql.questions,mysqladmin -uroot status | cut -f4 -d":" | cut -f1 -d"S"
```

mysql.questions to unikatowy identyfikator. Może to być dowolny prawidłowy identyfikator klucza, na przykład *queries*.

Przetestuj tę pozycję, używając agenta Zabbix z flagą `-t` (jeśli agent jest uruchamiany jako root, pamiętaj jednak, że po uruchomieniu jako demon może mieć inne uprawnienia):

```
zabbix_agentd -t mysql.questions
```

Krok 3

Przeładuj parametry użytkownika z pliku konfiguracyjnego, uruchamiając:

```
zabbix_agentd -R userparameter_reload
```

Możesz także ponownie uruchomić agent zamiast używać polecenia sterowania w czasie działania.

Przetestuj parametr za pomocą narzędzia `zabbix_get`.

Krok 4

Dodaj nową pozycję z `Key=mysql.questions` do monitorowanego hosta. Typ pozycji musi być ustawiony jako *Zabbix Agent* lub *Zabbix Agent (active)*.

Należy pamiętać, że typ zwracanych wartości musi być poprawnie ustawiony na serwerze Zabbix. W przeciwnym razie Zabbix ich nie zaakceptuje.

Liczniki wydajności systemu Windows

Przegląd

Możesz skutecznie monitorować liczniki wydajności systemu Windows za pomocą klucza `perf_counter[]`.

Na przykład:

```
perf_counter["\Processor(0)\Interrupts/sec"]
```

lub

```
perf_counter["\Processor(0)\Interrupts/sec", 10]
```

Więcej informacji na temat korzystania z tego klucza lub jego dostępnego wyłącznie w języku angielskim odpowiednika `perf_counter_en` znajdziesz w sekcji Klucze pozycji specyficzne dla systemu Windows.

Aby uzyskać pełną listę liczników wydajności dostępnych do monitorowania, możesz uruchomić:

```
typeperf -qx
```

Możesz również użyć wykrywania niskiego poziomu do wykrywania wielu instancji obiektów liczników wydajności systemu Windows i zautomatyzować tworzenie pozycji `perf_counter` dla obiektów wielu instancji.

Reprezentacja numeryczna

System Windows przechowuje numeryczne reprezentacje (indeksy) nazw obiektów i liczników wydajności. Zabbix obsługuje te reprezentacje numeryczne jako parametry kluczy pozycji `perf_counter`, `perf_counter_en` oraz parametrów konfiguracyjnych `PerfCounter`, `PerfCounterEn`.

Nie zaleca się jednak ich używania, chyba że można zagwarantować, że indeksy numeryczne odpowiadają prawidłowym ciągom znaków na określonych hostach. Jeśli potrzebujesz utworzyć przenośne pozycje, które działają na różnych hostach z różnymi zlokalizowanymi wersjami systemu Windows, możesz użyć klucza `perf_counter_en` lub parametru konfiguracyjnego `PerfCounterEn`, które umożliwiają używanie nazw angielskich niezależnie od ustawień regionalnych systemu.

Aby znaleźć odpowiedniki numeryczne, uruchom `regedit`, a następnie znajdź wpis *Counter* w `HKEY_LOCAL_MACHINE\SOFTWARE\Microsof NT\CurrentVersion\Perflib\009`.

Wpis rejestru zawiera informacje takie jak:

```
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
....
```

Tutaj można znaleźć odpowiednie liczby dla każdej części tekstowej licznika wydajności, na przykład w `\System\% Processor Time`:

```
System → 2
% Processor Time → 6
```

Następnie można użyć tych liczb do przedstawienia ścieżki w postaci numerycznej:

```
\2\6
```

Parametry liczników wydajności

Możesz wdrożyć niektóre parametry PerfCounter do monitorowania liczników wydajności systemu Windows.

Na przykład możesz dodać je do pliku konfiguracyjnego Zabbix agent:

```
PerfCounter=UserPerfCounter1,"\Memory\Page Reads/sec",30
lub
PerfCounter=UserPerfCounter2,"\4\24",30
```

Po skonfigurowaniu takich parametrów możesz po prostu użyć `UserPerfCounter1` lub `UserPerfCounter2` jako kluczy podczas tworzenia odpowiednich pozycji.

Pamiętaj, aby po wprowadzeniu zmian w pliku konfiguracyjnym ponownie uruchomić Zabbix agent.

Masowa aktualizacja

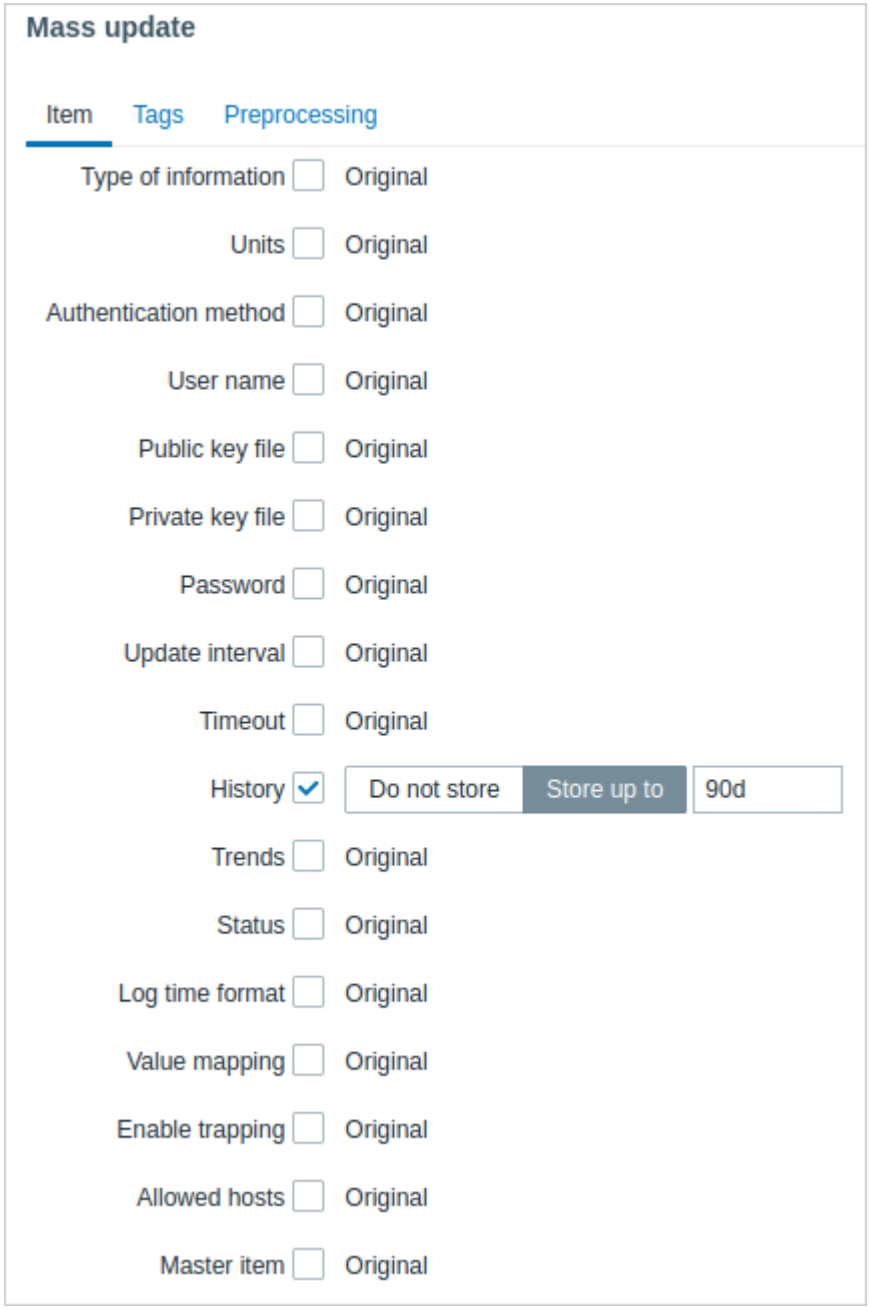
Przegląd

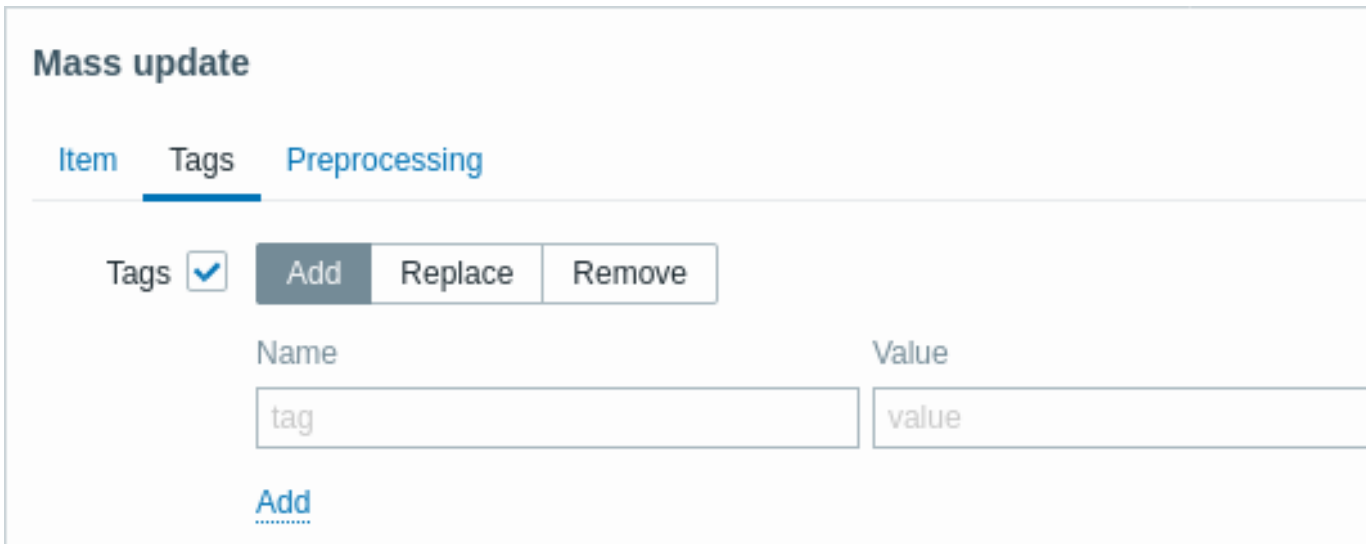
Czasami możesz chcieć zmienić jakiś atrybut dla wielu pozycji jednocześnie. Zamiast otwierać każdą pojedynczą pozycję do edycji, możesz w tym celu użyć funkcji masowej aktualizacji.

Aktualizacja zbiorcza

Aby zbiorczo zaktualizować niektóre pozycje, wykonaj następujące czynności:

- Zaznacz pola wyboru pozycji, które chcesz zaktualizować na liście
- Kliknij opcję *Aktualizacja zbiorcza* poniżej listy
- Przejdź do karty z wymaganymi atrybutami (*Pozycja*, *Znaczniki* lub *Wstępne przetwarzanie*)
- Zaznacz pola wyboru atrybutów, które chcesz zaktualizować
- Wprowadź nowe wartości atrybutów



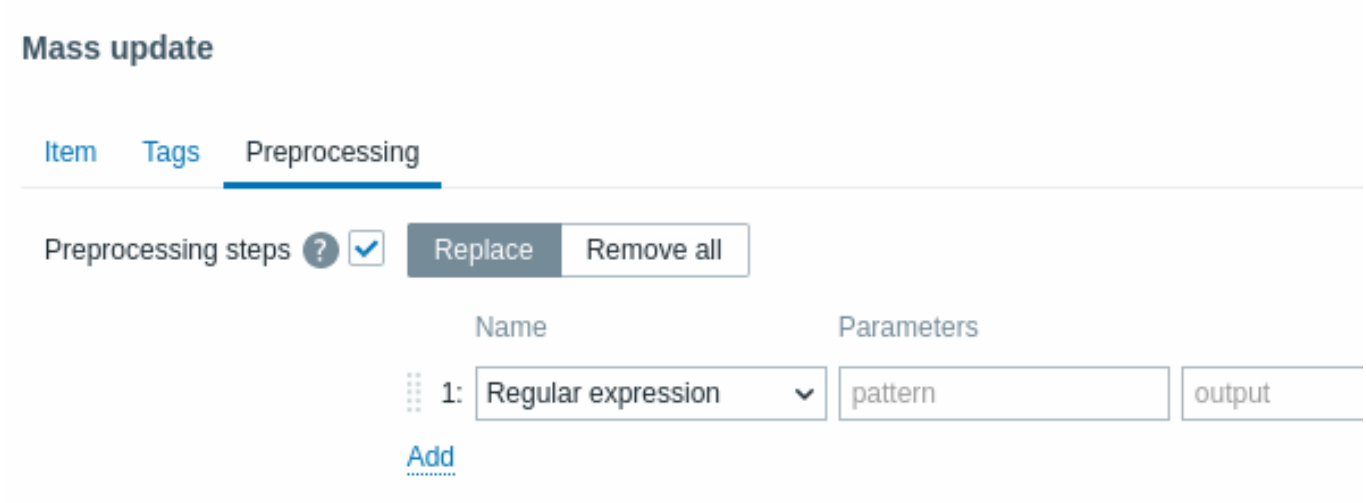


Opcja *Znaczniki* umożliwia:

- *Dodawanie* - dodanie określonych znaczników do pozycji (znaczniki o tej samej nazwie, ale z różnymi wartościami nie są uznawane za „duplikaty” i można je dodać do tej samej pozycji).
- *Zastępowanie* - usunięcie określonych znaczników i dodanie znaczników z nowymi wartościami

- *Usuwanie* - usunięcie określonych znaczników z pozycji

W znacznikach obsługiwane są makra użytkownika, makra {INVENTORY.*}, makra {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} i {HOST.ID}.



Zaznacz pole wyboru obok opcji *Kroki wstępnego przetwarzania*, aby aktywować zbiorczą aktualizację kroków wstępnego przetwarzania.

Zbiorcza aktualizacja *Wstępnego przetwarzania* umożliwia:

- *Zastępowanie* - zastąpienie istniejących kroków wstępnego przetwarzania w pozycjach krokami wstępnego przetwarzania określonymi poniżej
- *Usuwanie wszystkich* - usunięcie wszystkich kroków wstępnego przetwarzania z pozycji

Po zakończeniu kliknij przycisk *Aktualizuj*.

Mapowanie wartości

Przegląd

Mapowanie wartości umożliwia skonfigurowanie bardziej przyjaznej dla użytkownika reprezentacji otrzymanych wartości przy użyciu mapowań między wartościami numerycznymi/tekstowymi a ich reprezentacjami tekstowymi.

Na przykład, gdy wartość pozycji to „0” lub „1”, mapowania wartości mogą zostać użyte do przedstawienia tych wartości w bardziej przyjazny dla użytkownika sposób:

- 0 → Niedostępne
- 1 → Dostępne

Mapowania wartości dla typów kopii zapasowych danych mogą zostać skonfigurowane następująco:

- F → Pełna
- D → Różnicowa
- I → Przyrostowa

Zakresy wartości dla napięcia mogą zostać skonfigurowane następująco:

- <=209 → Niskie
- 210-230 → OK
- >=231 → Wysokie

Mapowanie wartości jest używane w frontendzie Zabbix oraz w powiadomieniach wysyłanych przez typy mediów.

Attention:
Podstawienie otrzymanej wartości skonfigurowaną reprezentacją jest wykonywane zarówno w frontendzie Zabbix, jak i na serwerze; jednak serwer obsługuje podstawienie tylko w następujących przypadkach:

- podczas wypełniania inwentarza hosta;
- podczas rozwijania obsługiwanych makr {ITEM.VALUE}, {ITEM.LASTVALUE}, {EVENT.OPDATA} oraz {EVENT.CAUSE.OPDATA}.

Mapowania wartości są konfigurowane w szablonach lub hostach. Po skonfigurowaniu są dostępne dla wszystkich pozycji w odpowiednim szablonie lub hoście. Podczas konfigurowania pozycji należy podać nazwę wcześniej skonfigurowanego mapowania wartości w parametrze *Mapowanie wartości*.

Note:
Nie ma dziedziczenia mapowań wartości — hosty i szablony nie dziedziczą mapowań wartości z podłączonych szablonów. Pozycje szablonu na hoście będą nadal używać mapowań wartości skonfigurowanych w szablonie.

Note:
Mapowania wartości mogą być używane z pozycjami mającymi typ informacji *Numeric (unsigned)*, *Numeric (float)* oraz *Character*.

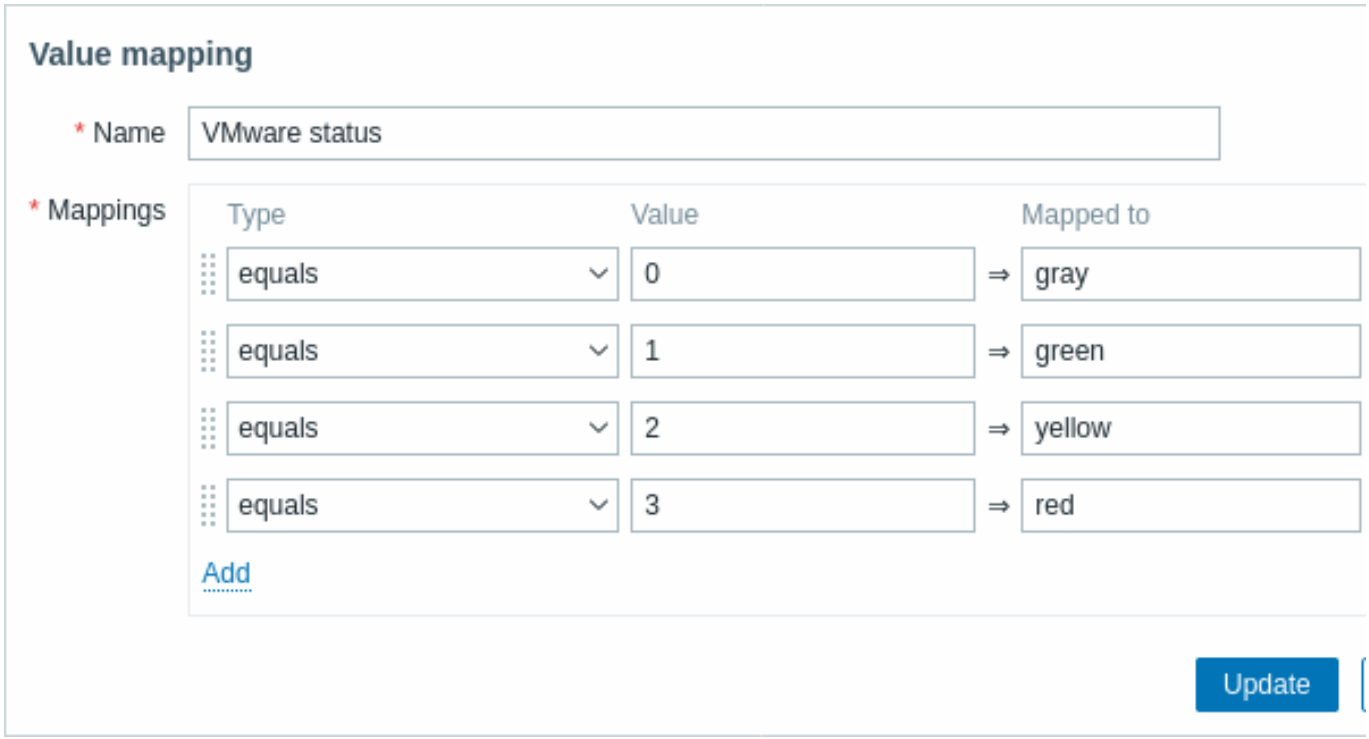
Mapowania wartości są eksportowane/importowane wraz z odpowiednim szablonem lub hostem. Mogą również być masowo aktualizowane przy użyciu formularzy masowej aktualizacji hosta i szablonu.

Konfiguracja

Aby skonfigurować mapowanie wartości, wykonaj następujące kroki:

1. Otwórz formularz konfiguracji host lub szablonu.

2. Na karcie *Mapowanie wartości* kliknij *Dodaj*, aby dodać nowe mapowanie wartości, lub kliknij nazwę istniejącego mapowania, aby je edytować.



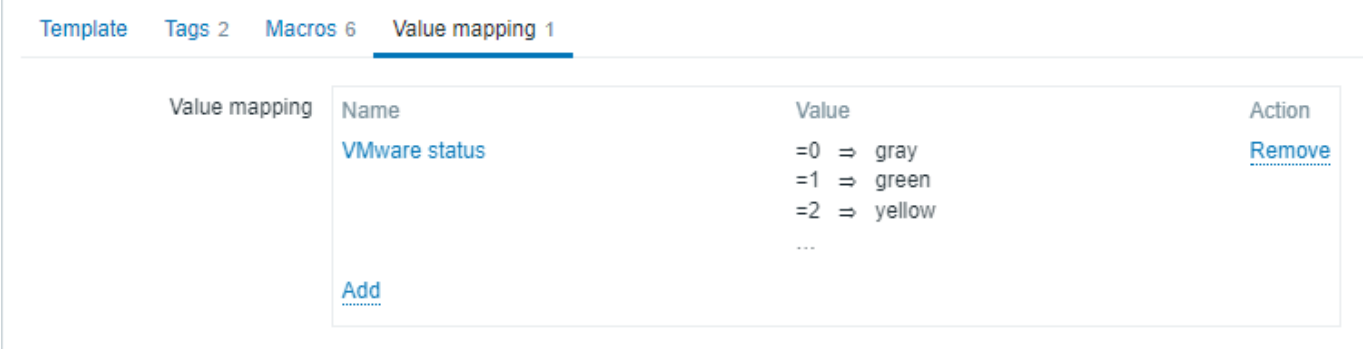
Parametry mapowania wartości:

Parametr	Opis
Nazwa	Unikalna nazwa zestawu mapowań wartości.
Mapowania	Poszczególne reguły mapowania wartości liczbowych/łańcuchowych na reprezentacje łańcuchowe. Mapowanie jest stosowane w kolejności reguł, którą można zmienić, przeciągając reguły.

Parametr	Opis
Typ	Typ mapowania: **equals** - mapowane będą wartości równe; **is greater than or equals** - mapowane będą wartości równe lub większe; **is less than or equals** - mapowane będą wartości równe lub mniejsze; **in range** - mapowane będą wartości z zakresu; zakres jest wyrażany jako <number1>-<number2> lub <number>; obsługiwanych jest wiele zakresów (na przykład 1-10,101-110,201); **regexp** - mapowane będą wartości odpowiadające wyrażeniu regularnemu (globalne wyrażenia regularne nie są obsługiwane); **default** - mapowane będą wszystkie pozostałe wartości, z wyjątkiem tych, dla których zdefiniowano określone mapowania. W przypadku mapowania zakresów obsługiwane są tylko liczbowe typy wartości (*is greater than or equals*, *is less than or equals*, *in range*).
Wartość	Wartość wejściowa (może zawierać zakres lub wyrażenie regularne, zależnie od typu mapowania).
Mapuj na	Reprezentacja łańcuchowa (do 64 znaków) dla wartości wejściowej.

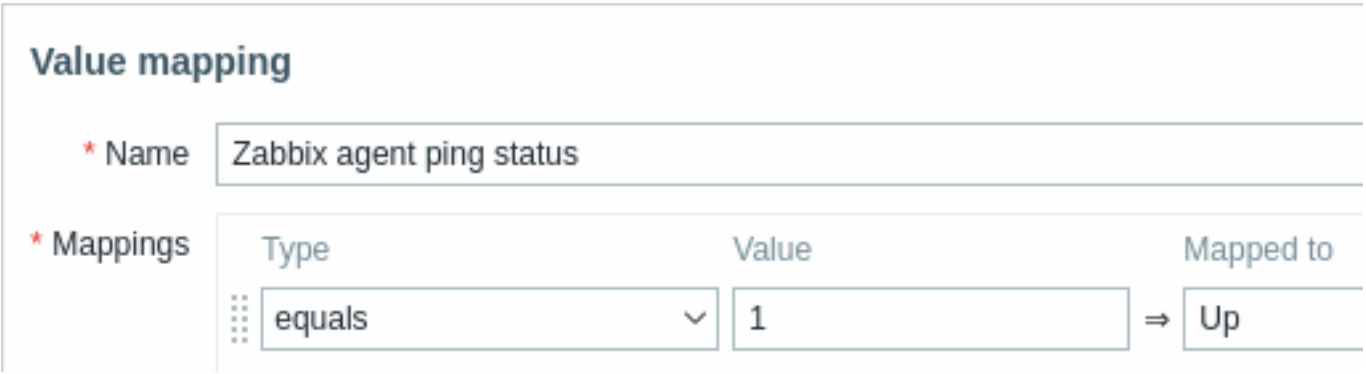
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Podczas wyświetlania mapowania wartości na liście widoczne są tylko trzy pierwsze mapowania, a trzy kropki wskazują, że istnieje więcej mapowań.



Przykład mapowania wartości

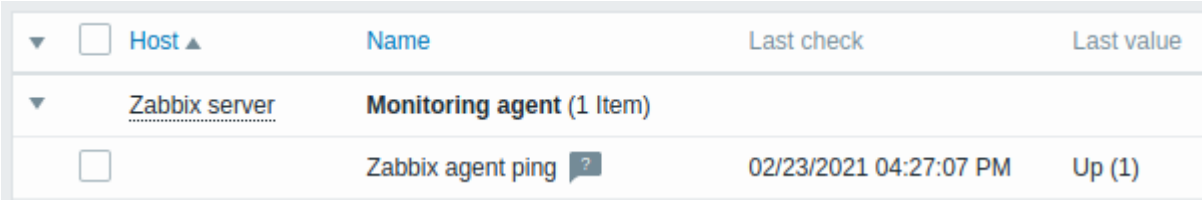
Jedna ze wstępnie zdefiniowanych pozycji agenta *Zabbix agent ping* używa mapowania wartości na poziomie szablonu „Zabbix agent ping status” do wyświetlania swoich wartości.



W formularzu konfiguracji pozycji można znaleźć odwołanie do tego mapowania wartości w polu *Mapowanie wartości*:

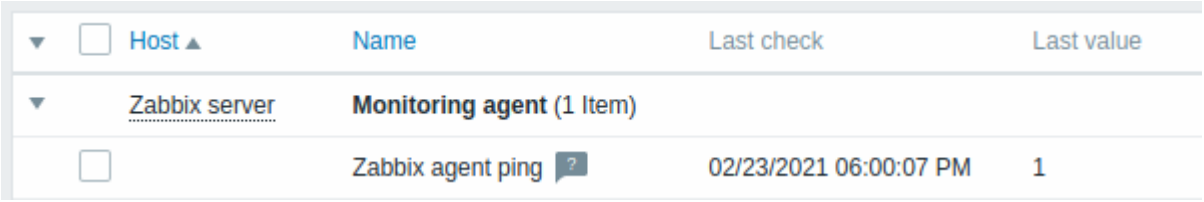


To mapowanie jest używane w sekcji *Monitoring* > *Latest data* do wyświetlania wartości „Up” (z wartością surową w nawiasach).



Note:
W sekcji *Latest data* wyświetlane wartości są skracane do 20 znaków. Jeśli używane jest mapowanie wartości, skrócenie nie jest stosowane do zmapowanej wartości, lecz tylko do wartości surowej (wyświetlanej w nawiasach).

Bez wstępnie zdefiniowanego mapowania wartości widoczna byłaby tylko wartość „1”, której zrozumienie może być trudne.



Kolejka

Przegląd

Kolejka wyświetla pozycje oczekujące na odświeżenie. Kolejka jest tylko **logiczną** reprezentacją danych. W Zabbix nie ma kolejki IPC ani żadnego innego mechanizmu kolejkowania.

Pozycje monitorowane przez proxy są również uwzględniane w kolejce — będą liczone jako oczekujące w okresie aktualizacji danych historycznych proxy.

W kolejce wyświetlane są tylko pozycje z zaplanowanymi czasami odświeżania. Oznacza to, że następujące typy pozycji są wykluczone z kolejki:

- aktywne pozycje log, logrt i event log agenta Zabbix
- pozycje SNMP trap
- pozycje trapper
- pozycje monitorowania WWW
- pozycje zależne

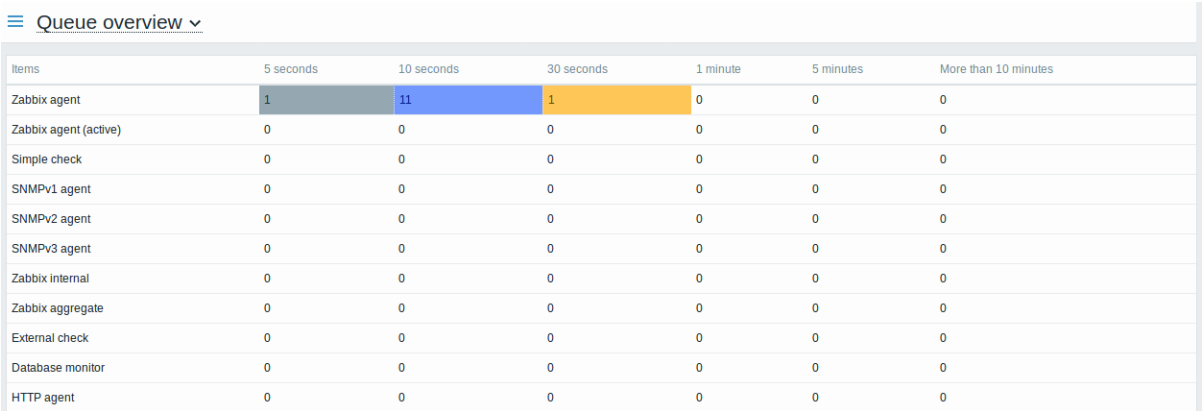
Statystyki pokazywane przez kolejkę są dobrym wskaźnikiem wydajności serwera Zabbix.

Kolejka jest pobierana bezpośrednio z serwera Zabbix przy użyciu protokołu JSON. Informacje są dostępne tylko wtedy, gdy serwer Zabbix jest uruchomiony.

Attention:
Pozycje nie są uwzględniane w kolejce, jeśli interfejs pozycji staje się niedostępny z powodu problemów z połączeniem lub nieprawidłowego działania agenta.

Odczytywanie kolejki

Aby odczytać kolejkę, przejdź do *Administracja* > *Kolejka* > *Przegląd kolejki*.



Przedstawiona tutaj sytuacja jest ogólnie „prawidłowa”, więc możemy założyć, że serwer działa poprawnie.

Kolejka pokazuje kilka pozycji oczekujących do 30 sekund. Dobrze byłoby wiedzieć, jakie to pozycje.

Aby to sprawdzić, wybierz *Szczegóły kolejki* z listy rozwijanej tytułu. Teraz możesz zobaczyć listę tych opóźnionych pozycji.

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

Dzięki tym szczegółom można ustalić, dlaczego te pozycje mogą być opóźnione.

Jedna lub dwie opóźnione pozycje nie muszą być powodem do niepokoju. Mogą zostać zaktualizowane w ciągu sekundy. Jeśli jednak widzisz wiele pozycji, których aktualizacja jest zbyt długo opóźniona, może to wskazywać na poważniejszy problem.

Zobacz także: Wyrównywanie stref czasowych podczas korzystania z interwałów harmonogramu.

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

Pozycja kolejki

Do monitorowania stanu kolejki w Zabbix można użyć specjalnej wewnętrznej pozycji `zabbix[queue,<from>,<to>]`. Zwróci ona liczbę pozycji opóźnionych o ustawiony czas. Więcej informacji można znaleźć w sekcji Wewnętrzne pozycje.

Pamięć podręczna wartości

Przegląd

Aby znacznie przyspieszyć obliczanie wyrażeń wyzwalaczy, pozycji obliczanych oraz niektórych makr, serwer Zabbix obsługuje opcję pamięci podręcznej wartości.

Ta pamięć podręczna w pamięci operacyjnej może być używana do uzyskiwania dostępu do danych historycznych zamiast wykonywania bezpośrednich wywołań SQL do bazy danych. Jeśli wartości historyczne nie są obecne w pamięci podręcznej, brakujące wartości są pobierane z bazy danych, a pamięć podręczna jest odpowiednio aktualizowana.

Wartości pozycji pozostają w pamięci podręcznej wartości do momentu, gdy:

- pozycja zostanie usunięta (wartości z pamięci podręcznej są usuwane po następnej synchronizacji konfiguracji);
- wartość pozycji znajdzie się poza zakresem czasu lub liczby określonym w wyrażeniu wyzwalacza/obliczanej pozycji (wartość z pamięci podręcznej jest usuwana po odebraniu nowej wartości);
- zakres czasu lub liczby określony w wyrażeniu wyzwalacza/obliczanej pozycji zostanie zmieniony tak, że do obliczeń wymagane będzie mniej danych (niepotrzebne wartości z pamięci podręcznej są usuwane po 24 godzinach).

Note:
Stan pamięci podręcznej wartości można obserwować za pomocą opcji runtime control serwera `diaginfo` (lub `diaginfo=valuecache`) oraz sprawdzając sekcję z informacjami diagnostycznymi pamięci podręcznej wartości. Może to być przydatne przy określaniu błędnie skonfigurowanych wyzwalaczy lub obliczanych pozycji.

Aby włączyć funkcję pamięci podręcznej wartości, można określić opcjonalny parametr `ValueCacheSize` w konfiguracji serwera Zabbix.

Do monitorowania pamięci podręcznej wartości obsługiwane są dwie pozycje wewnętrzne: `zabbix[vcache,buffer,<mode>]` i `zabbix[vcache,cache,<parameter>]`. Więcej informacji można znaleźć w sekcji dotyczącej pozycji wewnętrznych.

Wykonaj teraz

Przegląd

Opcja *Wykonaj teraz* umożliwia natychmiastowe wykonywanie kontroli pasywnych.

Pobieranie wartości pozycji w Zabbix odbywa się zgodnie ze skonfigurowanymi interwałami aktualizacji. Niektóre pozycje (np. reguły wykrywania niskopoziomowego) mają długie interwały i może być konieczne natychmiastowe pobranie nowej wartości — na przykład w celu wykrycia zmian w wykrywalnych zasobach.

Opcja *Wykonaj teraz* jest obsługiwana dla następujących typów pozycji:

- agent Zabbix (pasywny)
- Kontrola prosta
- agent SNMP (v1/v2/v3)
- Wewnętrzny Zabbix
- Kontrola zewnętrzna
- Monitor bazy danych
- agent HTTP
- agent IPMI
- agent SSH
- agent TELNET
- agent JMX
- Obliczana
- Pozycja zależna
- Skrypt
- Przeglądarka

Jeśli pozycja jest typu „Pozycja zależna”, jej pozycja nadrzędna również musi być jednym z powyższych typów.

Konfiguracja

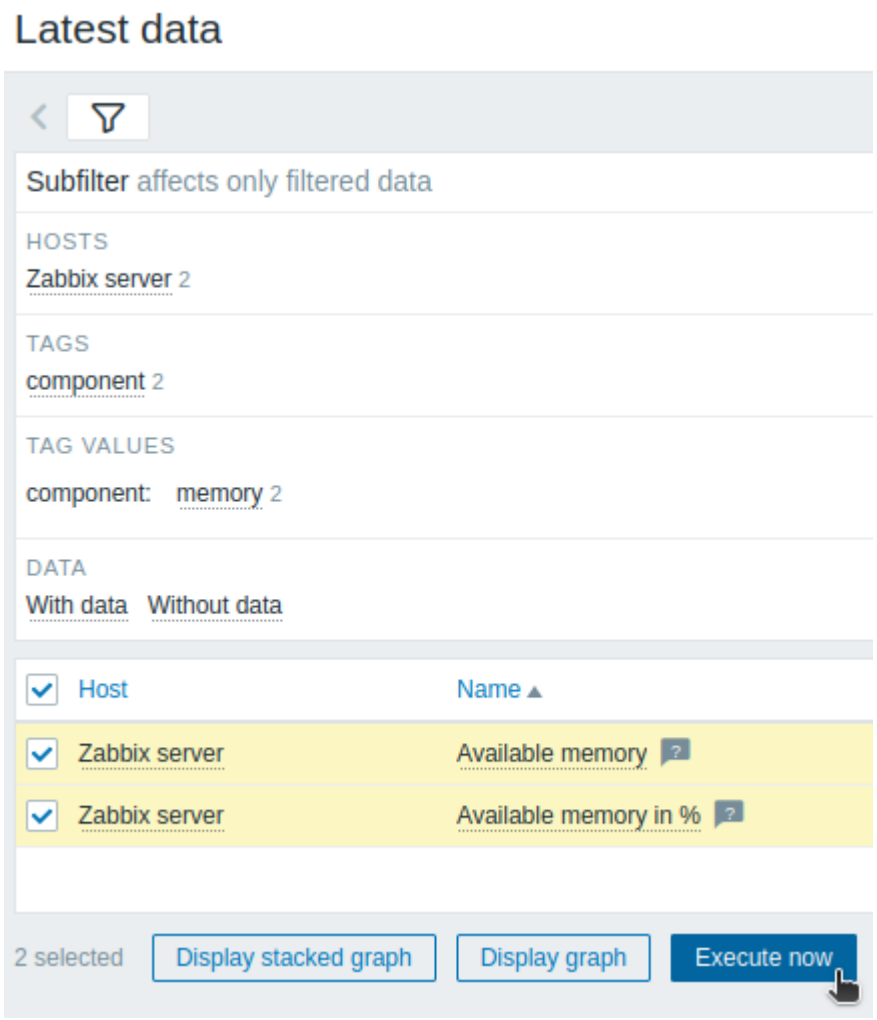
Możesz wykonać pasywne sprawdzenia z:

- *Monitoring* > *Latest data*
- *Data collection* > *Hosts* > *Items* lub *Discovery*
- menu pozycji
- Podczas edycji pozycji lub reguł niskopoziomowego wykrywania (w ich formularzach konfiguracji)

Attention:
Sprawdzenie musi istnieć w pamięci podręcznej konfiguracji, aby mogło zostać wykonane; zobacz także CacheUpdateFrequency. Pamięć podręczna **nie jest aktualizowana** przed wykonaniem sprawdzenia, więc najnowsze zmiany w konfiguracji pozycji lub reguły niskopoziomowego wykrywania nie zostaną uwzględnione. Aby przetestować pozycje lub reguły LLD, które są tworzone lub zostały właśnie utworzone, możesz zamiast tego użyć opcji Test.

Na przykład, aby wykonać pasywne sprawdzenie w *Monitoring* > *Latest data*:

1. Wybierz pozycje; możesz wykonać sprawdzenie dla wielu pozycji jednocześnie.
2. Kliknij *Execute now*.



Note:
W *Latest data* użytkownicy mogą *Execute now* dla hostów, do których mają uprawnienie *read-write* permission, albo uprawnienie *read* oraz włączoną dla swojej roli opcję *Invoke "Execute now" on read-only hosts* action.

Ograniczanie sprawdzeń agentów

Omówienie

Możesz kontrolować, których kluczy pozycji Zabbix agent lub agent 2 może używać albo których nie może używać podczas wykonywania sprawdzeń pozycji, zdalnych poleceń lub skryptów.

Aby to zrobić, użyj tych parametrów konfiguracji agenta, aby zdefiniować reguły zezwalania/odmawiania:

- `AllowKey=<pattern>`
- `DenyKey=<pattern>`
- `AllowKeyRegexp=<pattern>`
- `DenyKeyRegexp=<pattern>`

`<pattern>` musi zawierać pojedynczy klucz pozycji.
 W przypadku `AllowKey` i `DenyKey` parametr `<pattern>` obsługuje symbole wieloznaczne (`*`). Symbol wieloznaczny dopasowuje dowolną liczbę dowolnych znaków na swojej pozycji i może być używany do dopasowywania kluczy pozycji lub parametrów (np. `vfs.file.*[*]`).
 W przypadku `AllowKeyRegexp` i `DenyKeyRegexp` parametr `<pattern>` obsługuje wyrażenia regularne. Wzorzec wyrażenia regularnego może być używany do dopasowywania kluczy pozycji i ich ciągów parametrów. Zobacz także przykłady poprawnego escapowania.
 Należy pamiętać, że Zabbix agent i Zabbix agent 2 mogą obsługiwać różną składnię wyrażeń regularnych.

Attention:
Aby poprawić bezpieczeństwo, zaleca się używanie dokładnych kluczy pozycji zamiast symboli wieloznacznych dla `AllowKey` i `DenyKey`. Zaleca się również używanie jak najbardziej szczegółowych wzorców wyrażeń regularnych dla `AllowKeyRegexp` i `DenyKeyRegexp`. Szczegóły znajdziesz w sekcji Zabezpieczanie reguł zezwalania/odmawiania.

W przeciwieństwie do innych parametrów konfiguracji agenta możesz określić nieograniczoną liczbę parametrów AllowKey, DenyKey, AllowKeyRegexp lub DenyKeyRegexp.

Ważne uwagi

- Wszystkie pozycje system.run są domyślnie wyłączone (nawet gdy DenyKey i DenyKeyRegexp są puste), tak jakby ustawiono DenyKey=system.run[*] lub DenyKeyRegexp=^system\.run\[.*\]$ jako ostatnią regułę. Z tego powodu można zezwolić na określone pozycje system.run bez jawnego blokowania innych pozycji system.run.
- Jeśli to możliwe, używaj AllowKey/AllowKeyRegexp, aby zezwalać tylko na wymagane pozycje i blokować wszystko inne. Niektóre klucze mogą zostać wykorzystane do odczytu niezamierzonych plików poprzez przejście po ścieżce (na przykład vfs.file.contents["../../../../etc/passwd"]), a nowe wersje agenta Zabbix mogą wprowadzać klucze nieobjęte regułami DenyKey/DenyKeyRegexp.
- Konfiguracja AllowKey, DenyKey, AllowKeyRegexp i DenyKeyRegexp nie wpływa na parametry agenta HostnameItem, HostMetadataItem ani HostInterfaceItem.
- Zablokowane pozycje stają się nieobsługiwane bez żadnych wskazówek ani komunikatów o błędach; na przykład:
  - Parametr wiersza poleceń agenta Zabbix --print (-p) nie będzie wyświetlał zablokowanych kluczy pozycji.
  - Parametr wiersza poleceń agenta Zabbix --test (-t) zwróci dla zablokowanych kluczy pozycji komunikat "Unsupported item key.".
  - Gdy rejestrowanie jest włączone (LogRemoteCommands=1), plik dziennika agenta Zabbix nie będzie rejestrował zablokowanych zdalnych poleceń.

Kolejność reguł allow/deny

Możesz określić nieograniczoną liczbę reguł AllowKey, DenyKey, AllowKeyRegexp lub DenyKeyRegexp, jednak ich kolejność ma znaczenie.

- Reguły są oceniane jedna po drugiej, od góry do dołu.
- Gdy klucz pozycji pasuje do reguły, jest ona albo dozwolona, albo odrzucona, a ocena reguł zostaje zatrzymana.

Na przykład podczas oceny vfs.file.contents[/etc/passwd] reguły są przetwarzane w następujący sposób:

```
AllowKey=vfs.file.contents[/tmp/app.log]    # Wzorzec klucza pozycji nie pasuje, agent przechodzi do nastę
AllowKey=vfs.file.contents[/etc/passwd]     # Wzorzec klucza pozycji pasuje; agent zezwala na sprawdzenie
DenyKey=vfs.file.*[*]                       # Agent ignoruje regułę, ponieważ ocena została zatrzymana.
```

Poniższa kolejność reguł spowoduje odrzucenie sprawdzenia pozycji:

```
DenyKey=vfs.file.*[*]                       # Wzorzec klucza pozycji pasuje; agent odrzuca sprawdzenie poz
AllowKey=vfs.file.contents[/etc/passwd]     # Agent ignoruje regułę, ponieważ ocena została zatrzymana.
AllowKey=vfs.file.contents[/tmp/app.log]    # Agent ignoruje regułę, ponieważ ocena została zatrzymana.
```

Przykłady

Poniższe przykłady pokazują typowe wzorce konfiguracji dla AllowKey, DenyKey, AllowKeyRegexp i DenyKeyRegexp.

Zezwalanie na określone sprawdzenia i polecenia

Zezwalaj tylko na dwa sprawdzenia pozycji vfs.file i dwa polecenia system.run:

używając AllowKey i 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.*[*]
```

używając AllowKeyRegexp i DenyKeyRegexp:

```
AllowKeyRegexp=^vfs\.file\.(contents|size)\[/tmp/app\.log\]$
AllowKeyRegexp=^system\.run\[/usr/bin/(uptime|df -h /)\]$
DenyKeyRegexp=^vfs\.file\..*\[.*\]$
```

Note:
Ustawienie DenyKey=system.run[*] lub DenyKeyRegexp=^system\.run\[.*\]$ jest niepotrzebne, ponieważ wszystkie pozostałe polecenia system.run są domyślnie odrzucane.

Zezwalanie na skrypty

Zezwól agentowi Zabbix na wykonywanie skryptów na hostach za pomocą wszystkich dostępnych metod:

- Skrypty globalne, które mogą być wykonywane w frontend lub przez API (ta metoda zawsze używa klucza `system.run[myscript.sh]`)
- Polecenia zdalne z operacji akcji (ta metoda zawsze używa klucza `system.run[myscript.sh,nowait]`)
- Pozycje `system.run` agenta Zabbix ze skryptem, na przykład:
  - `system.run[myscript.sh]`
  - `system.run[myscript.sh,wait]`
  - `system.run[myscript.sh,nowait]`

```
AllowKey=system.run[myscript.sh,*]
```

lub

```
AllowKeyRegexp=^system\.run\[myscript\.sh,.*\]$
```

Aby kontrolować parametr wait/nowait, musisz ustawić inną regułę. Na przykład możesz zezwolić tylko na pozycje `system.run[myscript.sh,wait]`, wykluczając w ten sposób inne metody:

```
AllowKey=system.run[myscript.sh,wait]
```

lub

```
AllowKeyRegexp=^system\.run\[myscript\.sh,wait\]$
```

Zabezpieczanie reguł allow/deny

Te przykłady pokazują, jak zabezpieczyć zbyt liberalne reguły `AllowKey`/`AllowKeyRegexp` lub `DenyKey`/`DenyKeyRegexp`.

Rozważ następujące reguły:

z użyciem `AllowKey` i `DenyKey`:

```
AllowKey=system.run["C:\Program^ Files\Zabbix^ Agent^ 2\scripts\test.bat*"]
DenyKey=vfs.file.*
DenyKey=system.cpu.load[*]
```

z użyciem `AllowKeyRegexp` i `DenyKeyRegexp`:

```
AllowKeyRegexp=^system\.run\["C:\\Program\sFiles\\Zabbix\sAgent\s2\\scripts\\test\.bat.*"\]$
DenyKeyRegexp=^vfs\.file\..*$
DenyKeyRegexp=^system\.cpu\.load\[.*\]$
```

Note:
W systemie Windows należy uciekać spacje w ścieżce za pomocą daszka (^) dla `AllowKey` lub `DenyKey` oraz `\s` dla `AllowKeyRegexp` lub `DenyKeyRegexp`.

Te reguły zawierają wieloznacznik (*) (lub `.*` w przypadku wzorców opartych na wyrażeniach regularnych), który może zostać użyty w niepożądany sposób:

- Skrypt `test.bat` może zostać uruchomiony z dowolnymi argumentami, w tym niezamierzonymi.
- Wzorzec `vfs.file.*` (`^vfs\.file\..*$`) dopasowuje klucze pozycji zarówno z parametrami, jak i bez nich; jednak wszystkie pozycje `vfs.file` wymagają parametrów.
- Wzorzec `system.cpu.load[*]` (`^system\.cpu\.load\[.*\]$`) dopasowuje tylko klucze pozycji z parametrami; jednak pozycje `system.cpu.load` nie wymagają parametrów.

Aby zabezpieczyć te reguły, należy jawnie zezwolić na uruchamianie `test.bat` tylko z określonymi argumentami oraz zablokować poprawne wzorce kluczy pozycji; na przykład:

z użyciem `AllowKey` i `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[*]
```

z użyciem `AllowKeyRegexp` i `DenyKeyRegexp`:

```
AllowKeyRegexp=^system\.run\["C:\\Program\sFiles\\Zabbix\sAgent\s2\\scripts\\test\.bat (status|version)"\]
DenyKeyRegexp=^vfs\.file\..+\[.*\]$
DenyKeyRegexp=^system\.cpu\.load(\[.*\])?$
```

Reguły można przetestować, uruchamiając następujące polecenia, które zwrócą 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]
```

Przykłady wzorców

Poniższe tabele pokazują, jak dopasowywane są wzorce kluczy pozycji:

- Klucz pasuje do wzorca tylko wtedy, gdy spełnia **wszystkie** warunki w kolumnie *Matches*.
- Parametry muszą być w całości ujęte w nawiasy kwadratowe (np. `vfs.file.contents[*` i `vfs.file.contents*utf8]` są nieprawidłowymi wzorcami).
- Przykłady ilustrują wyłącznie dopasowywanie wzorców; rzeczywiste pozycje wymagają parametrów.

Dla `AllowKey` i `DenyKey`:

Pattern	Matches	Examples
`*`	Dowolny klucz z parametrami lub bez	

`vfs.file.*`	Klucz zaczyna się od `vfs.file.`
	Z parametrami lub bez
		Matches:
		`vfs.file.size`
		`vfs.file.contents`
		`vfs.file.contents[]`
		`vfs.file.size[/var/log/app.log]`

`vfs.*.contents`	Klucz zaczyna się od `vfs.`
	Klucz kończy się na `.contents`
	Bez parametrów
		Matches:
		`vfs..contents`
		`vfs.mount.point.file.contents`
		
		Does not match:
		`vfs.contents`
		`vfs.file.contents[]`

`vfs.file.*[*]`	Klucz zaczyna się od `vfs.file.`
	Dowolne lub puste parametry
		Matches
		`vfs.file.get.custom[]`
		`vfs.file.size[/var/log/app.log, utf8]`
		
		Does not match:
		`vfs.file.get.custom`

`vfs.file.contents`	Klucz to `vfs.file.contents`
	Bez parametrów
		Matches:
		`vfs.file.contents`
		
		Does not match:
		`vfs.file.contents[/etc/passwd]`

`vfs.file.contents[]`	Klucz to `vfs.file.contents[]`
	Puste parametry
		Matches:
		`vfs.file.contents[]`
		
		Does not match:
		`vfs.file.contents`

`vfs.file.contents[*]`	Klucz to `vfs.file.contents`
	Dowolne lub puste parametry
		Matches:
		`vfs.file.contents[/path/to/file]`
		
		Does not match:
		`vfs.file.contents`

`vfs.file.contents[/etc/passwd,*]`	Klucz to `vfs.file.contents`
	Pierwszy parametr to `/etc/passwd`
	Dowolny lub pusty drugi parametr
		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*]	Klucz to vfs.file.contents Dowolne parametry, co najmniej jeden zawiera passwd	Matches: vfs.file.contents[/etc/passwd] vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8]
vfs.file.contents[*passwd*,*]	Klucz to vfs.file.contents Pierwszy parametr zawiera passwd Dowolny lub pusty drugi parametr	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]	Klucz to vfs.file.contents Pierwszy parametr to /etc/passwd Drugi parametr to utf8	Matches: vfs.file.contents[/etc/passwd,utf8] Does not match: vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf16]

Dla AllowKeyRegexp i DenyKeyRegexp:

Pattern	Matches	Examples
^.*$	Dowolny klucz z parametrami lub bez	
^vfs\..*$	Klucz zaczyna się od vfs.file. Z parametrami lub bez	Matches: vfs.file.size vfs.file.contents vfs.file.contents[] vfs.file.size[/var/log/app.log]
^vfs\..*\.contents$	Klucz zaczyna się od vfs. Klucz kończy się na .contents Bez parametrów	Matches: vfs..contents vfs.mount.point.file.contents Does not match: vfs.contents vfs.file.contents[]
^vfs\.file\..*\[.*\]$	Klucz zaczyna się od vfs.file. Dowolne lub puste parametry	Matches vfs.file.get.custom[] vfs.file.size[/var/log/app.log, utf8] Does not match: vfs.file.get.custom
^vfs\.file\.contents$	Klucz to vfs.file.contents Bez parametrów	Matches: vfs.file.contents Does not match: vfs.file.contents[/etc/passwd]
^vfs\.file\.contents\[\]$	Klucz to vfs.file.contents[] Puste parametry	Matches: vfs.file.contents[] Does not match: vfs.file.contents
^vfs\.file\.contents\[.*\]$	Klucz to vfs.file.contents Dowolne lub puste parametry	Matches: vfs.file.contents[/path/to/file] Does not match: vfs.file.contents

Pattern	Matches	Examples
^vfs\.file\.contents\[/etc/passwd,?\]$	Klucz to vfs.file.contents Pierwszy parametr to /etc/passwd Dowolny lub pusty drugi parametr	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\[.*passwd.*\]$	Klucz to vfs.file.contents Dowolne parametry, co najmniej jeden zawiera passwd	Matches: vfs.file.contents[/etc/passwd] vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8]
^vfs\.file\.contents\[.*passwd.*,?\]$	Klucz to vfs.file.contents Pierwszy parametr zawiera passwd Dowolny lub pusty drugi parametr	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\[/etc/passwd,(utf8|windows-1252)\]$	Klucz to vfs.file.contents Pierwszy parametr to /etc/passwd Drugi parametr to utf8 lub 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]

Wykrywanie problemów za pomocą wyzwalaczy

Wykrywanie problemów w Zabbix opiera się na **wyzwalaczach**.

Wyzwalacz opisuje warunek problemu, umożliwiając Zabbix reakcję, jeśli problem wystąpi.

Pełny proces wykrywania problemów wygląda następująco:

1. Upewnij się, że masz pozycję zbierającą dane.
2. Zdefiniuj wyzwalacz dla tej pozycji. Bez wyzwalacza Zabbix tylko zbiera dane, ale nigdy nie reaguje na nie.
3. Obserwuj wykryte problemy:
   - W *Monitoring* > *Problems*
   - W widżecie *Problems*
4. Odbieraj powiadomienia, wykonuj zdalne polecenia (jeśli zostały zdefiniowane).

Należy pamiętać, że oficjalne szablony Zabbix zawierają predefiniowane wyzwalacze.
Zobacz przewodnik dotyczący zastosowania szablonu do monitorowania Linux.

Czym jest „wyzwalacz”?

Wyzwalacz opisuje *warunek problemu* za pomocą wyrażenia wyzwalacza.

Warunek problemu, gdy wystąpi w rzeczywistości, oznacza, że wyzwalacz „uruchamia się” (w Zabbix tworzony jest problem).

Warunek problemu może być oparty na najnowszej wartości, wartości średniej, wykrytym ciągu znaków i wielu innych funkcjach.
Wyzwalaczy nie można tworzyć dla pozycji zwracających wartości o typie danych binary lub JSON.

Proste wyrażenie wyzwalacza oblicza funkcję (na przykład `max()` dla wartości maksymalnej) dla danych określonych w nawiasach (zwykle pozycja danych i okno czasowe), a następnie porównuje wynik tego obliczenia z progiem.

Na przykład:

```
max(/host/vfs.fs.size[/,free],5m)<10G
```

Ten wyzwalacz uruchamia się, jeśli wolne miejsce na dysku, mierzone przez 5 minut, stale utrzymuje się poniżej 10 GB.

Wyzwalacz może mieć następujące stany:

Stan wyzwalacza	Opis	W Zabbix
OK	Wyrażenie wyzwalacza zostało obliczone jako „0” (lub FALSE)	Nie utworzono problemu Problem został rozwiązany (jeśli istniał)
Problem	Wyrażenie wyzwalacza zostało obliczone jako „1” (lub TRUE)	Utworzono problem

Czasami obliczenie wyrażenia wyzwalacza nie jest z jakiegoś powodu możliwe. Zobacz Nieznany stan wyrażenia.

Inne materiały:

- Przykłady wyzwalaczy
- Konfigurowanie wyzwalacza
- Najlepsze praktyki

Częstotliwość obliczeń

Wyzwalacz jest przeliczany za każdym razem, gdy serwer Zabbix otrzyma **nową wartość**, która jest częścią wyrażenia. Po otrzymaniu nowej wartości każda funkcja uwzględniona w wyrażeniu jest przeliczana (nie tylko ta, która otrzymała nową wartość).

Dodatkowo wyzwalacz jest przeliczany co 30 sekund (przez history syncer), jeśli wyrażenie zawiera:

- funkcje daty i czasu
- funkcję nodata()

Wyzwalacze zawierające wyłącznie funkcje trend są oceniane raz na najmniejszy okres czasu w wyrażeniu. Podczas gdy wiele funkcji wyzwalaczy używa danych historii wartości pozycji, funkcje trendów używają danych trendów.

Przykłady wyzwalaczy

Ta strona zawiera zbiór przykładów wyzwalaczy.

Przykłady są uporządkowane według przypadku użycia:

- Niedostępny host
- Niedostępny proxy
- Niedostępny klaster SMTP
- Niedostępny host
- Nieoczekiwany restart
- Zmiany w ważnych plikach
- Zmiany w zapytaniu DNS
- Niezgodne oprogramowanie na różnych hostach
- Niesynchronizowane zegary
- Nieaktualny agent
- Fluktuujący interfejs
- Duży ruch przychodzący
- Mało wolnego miejsca na dysku
- Mało wolnego miejsca na dysku (dynamiczny próg)
- Duże obciążenie CPU
- Porównanie obciążenia CPU
- Długoterminowe porównanie obciążenia CPU
- Łańcuch zawierający znaki specjalne

Wykrywanie niedostępnego hosta

```
max(/host/zabbix[host,agent,available],5m)=0
```

Ten wyzwalacz uruchamia się, jeśli agent Zabbix na hoście jest niedostępny przez 5 minut.
 Funkcja: `max`
 Pozycja danych: `zabbix[host,agent,available]`

Alternatywnie:

```
nodata(/host/agent.ping,5m)=1
```

Ten wyzwalacz uruchamia się, jeśli przez 5 minut nie odebrano żadnych danych od agenta Zabbix.
 Funkcja: `nodata`
 Pozycja danych: `agent.ping`

Wykrywanie niedostępnego proxy

```
fuzzytime(/host/zabbix[proxy,{$PROXY_NAME},lastaccess],1m)=0
```

Ten wyzwalacz uruchamia się, jeśli dane proxy Zabbix są opóźnione względem czasu serwera Zabbix o 1 minutę.
 Funkcja: fuzzytime
 Pozycja danych: zabbix[proxy,{$PROXY_NAME},lastaccess]

Wykrywanie niedostępnego klastra SMTP

```
last(/smtp1.example.com/net.tcp.service[smtp])=0 and last(/smtp2.example.com/net.tcp.service[smtp])=0
```

Ten wyzwalacz uruchamia się, jeśli oba serwery SMTP są niedostępne.
 Funkcja: last
 Pozycja danych: net.tcp.service

Wykrywanie nieosiągalnego hosta

```
count(/host/icmpping,30m,,"0")>5
```

Ten wyzwalacz uruchamia się, jeśli host jest nieosiągalny przez ping więcej niż 5 razy w ciągu ostatnich 30 minut.
 Funkcja: count
 Pozycja danych: icmpping

Wykrywanie nieoczekiwanego restartu

```
change(/host/system.uptime)<0
```

Ten wyzwalacz uruchamia się, jeśli zostanie wykryta ujemna zmiana wartości czasu działania systemu (co wskazuje na restart).
 Funkcja: change
 Pozycja danych: system.uptime

Wykrywanie zmian w ważnych plikach

```
last(/host/vfs.file.cksum[/etc/passwd],#1)<>last(/host/vfs.file.cksum[/etc/passwd],#2)
```

Ten wyzwalacz uruchamia się, jeśli plik /etc/passwd został zmieniony. Wyrażenie jest prawdziwe, gdy poprzednia suma kontrolna /etc/passwd różni się od najnowszej. Podobne wyrażenia mogą być przydatne do monitorowania zmian w ważnych plikach, takich jak /etc/passwd, /etc/inetd.conf, /kernel itd.
 Funkcja: last
 Pozycja danych: vfs.file.cksum

Wykrywanie zmian w zapytaniu DNS

```
last(/Zabbix server/net.dns.record[192.0.2.1,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1])<>"{$WEBSITE
```

Zwróć uwagę na cudzysłowy wokół drugiego operandu.

Ten wyzwalacz uruchamia się, jeśli wynik zapytania nie jest równy temu, co zwykle zwraca:

```
example.com             MX      0 mail.example.com
```

Funkcja: last
 Pozycja danych: net.dns.record[192.0.2.1,{$WEBSITE_NAME},{$DNS_RESOURCE_RECORD_TYPE},2,1], z makrami zdefiniowanymi jako:

```
{$WEBSITE_NAME} = example.com
{$DNS_RESOURCE_RECORD_TYPE} = MX
```

Wykrywanie niezgodnego oprogramowania na różnych hostach

```
last(/host/vfs.file.contents[/etc/os-release])<>last(/host2/vfs.file.contents[/etc/os-release])
```

Ten wyzwalacz uruchamia się, jeśli wersja Ubuntu jest różna na różnych hostach. Zwróć uwagę, że operandami są tutaj funkcje zwracające ciągi znaków.
 Funkcja: last
 Pozycja danych: vfs.file.contents

Wykrywanie niesynchronizowanych zegarów

```
fuzzytime(/host/system.localtime,10s)=0
```

Wyzwalacz uruchamia się, jeśli czas lokalny klienta i czas serwera Zabbix różnią się o więcej niż 10 sekund.
 Funkcja: fuzzytime
 Pozycja danych: system.localtime

Należy pamiętać, że system.localtime musi być skonfigurowana jako passive check dla agenta Zabbix; w przypadku Zabbix agent 2 może być skonfigurowana jako check aktywny.

Wykrywanie nieaktualnego agent

```
find(/host/agent.version,,"like","beta")=1
```

Ten wyzwalacz uruchamia się, jeśli Zabbix agent ma wersję beta. Zabbix agent wymaga aktualizacji.
 Funkcja: find
 Pozycja danych: agent.version

Wykrywanie niestabilnego interfejsu

```
changecount(/host/vfs.file.contents["/sys/class/net/eth0/operstate"],1h)>5
```

Ten wyzwalacz uruchamia się, jeśli stan operacyjny (up/down/unknown) interfejsu eth0 zmienił się więcej niż 5 razy w ciągu godziny.
 Funkcja: changecount
 Pozycja danych: vfs.file.contents

Wykrywanie wysokiego ruchu przychodzącego

```
min(/host/net.if.in[eth0,bytes],5m)>100K
```

Ten wyzwalacz uruchamia się, jeśli liczba odebranych bajtów na eth0 w ciągu ostatnich pięciu minut była zawsze większa niż 100 kilobajtów. Prawdopodobnie ktoś pobiera duży plik.
 Funkcja: min
 Pozycja danych: net.if.in[eth0,bytes]

Wykrywanie małej ilości wolnego miejsca na dysku

```
max(/host/vfs.fs.size[/,free],5m)<10G
```

Wyzwalacz uruchamia się, jeśli ilość wolnego miejsca na dysku jest stale (przez 5 minut) poniżej 10 GB.

Można również zdefiniować wyrażenie odzyskiwania:

```
min(/host/vfs.fs.size[/,free],10m)>40G
```

Problem zostaje **rozwiązany** dopiero wtedy, gdy ilość wolnego miejsca na dysku jest stale (przez 10 minut) powyżej 40 GB.
 Funkcja: min
 Pozycja danych: vfs.fs.size

Wykrywanie małej ilości wolnego miejsca na dysku (próg dynamiczny)

```
last(/template/hrStorageFree[{#SNMPVALUE}])<last(/template/hrStorageSize[{#SNMPVALUE}])*0.1
```

Wyzwalacz uruchamia się, jeśli ilość wolnego miejsca w pamięci masowej (w jednostkach alokacji) spadnie poniżej 10 procent. Zwróć uwagę na użycie wartości innej pozycji do uzyskania adaptacyjnego progu wyzwalacza, mającego zastosowanie do wykrytej pamięci masowej o różnej wielkości. Funkcja: last

Wykrywanie wysokiego obciążenia CPU

```
last(/host/system.cpu.load[all,avg1])>5
```

Wyzwalacz uruchamia się, gdy średnie obciążenie procesora przekracza 5 przez jedną minutę.

Warianty:

```
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
```

Takie wyzwalacze analizują dane z 5 minut i uruchamiają się tylko wtedy, gdy obciążenie CPU nigdy nie spada poniżej 2. Dodatkowo te wyzwalacze uruchamiają się:

- tylko w nocy (00:00-06:00)
- o dowolnej porze z wyjątkiem 2 godzin podczas zmiany tygodnia (niedziela, 23:00 - poniedziałek, 01:00)

```
(last(/host/system.cpu.load[all,avg1])>5) + (last(/host2/system.cpu.load[all,avg1])>5) + (last(/host3/syst
```

Ten wyzwalacz uruchamia się, jeśli obciążenie procesora jest zbyt wysokie na co najmniej dwóch z trzech hostów.
 Funkcje: last, min, dayofweek, time
 Pozycja danych: system.cpu.load

Porównanie obciążeń CPU

```
avg(/Zabbix server/system.cpu.load,1h)/avg(/Zabbix server/system.cpu.load,1h:now-1d)>2
```

Wyzwalacz uruchamia się, jeśli średnie obciążenie dzisiaj przekracza średnie obciążenie z tej samej godziny wczoraj (z użyciem przesunięcia czasu now-1d) ponad dwukrotnie.
 Funkcja: avg
 Pozycja danych: system.cpu.load

Porównanie długoterminowego obciążenia CPU

```
trendavg(/host/system.cpu.load,1M:now/M)>1.1*trendavg(/host/system.cpu.load,1M:now/M-1M)
```

Ten wyzwalacz uruchamia się, jeśli obciążenie CPU na hoście wzrosło w ostatnim miesiącu o więcej niż 10%
 Funkcja: trendavg
 Pozycja danych: system.cpu.load

Możesz również użyć pola Nazwa zdarzenia w konfiguracji wyzwalacza, aby zbudować zrozumiały komunikat alertu, na przykład aby otrzymać coś takiego

```
"Obciążenie serwera Exchange wzrosło o 24% w lipcu (0.69) w porównaniu do czerwca (0.56)"
```

nazwa zdarzenia musi być zdefiniowana jako:

```
Load of {HOST.HOST} server increased by {{?100*trendavg(//system.cpu.load,1M:now/M)/trendavg(//system.cpu.
```

Przy tego rodzaju problemie przydatne jest również zezwolenie na ręczne zamykanie w konfiguracji wyzwalacza.

Wykrywanie ciągu zawierającego znaki specjalne

```
last(/host/vfs.file.contents[/tmp/hello])={$HELLO_MACRO}
```

Wyzwalacz uruchamia się, jeśli zawartość `/tmp/hello` jest równa ciągowi zdefiniowanemu w `{$HELLO_MACRO}`:

```
{$HELLO_MACRO} = \" //hello ?\"
```

Alternatywnie można porównać bezpośrednio do ciągu:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])="\\\" //hello ?\\\""
```

Zwróć uwagę, że znaki specjalne (\\ i ") są poprzedzone znakiem ucieczki, gdy ciąg jest porównywany bezpośrednio.

Funkcja: `last`
 Pozycja danych: `vfs.file.contents`

Konfigurowanie wyzwalacza

Ta strona opisuje, jak skonfigurować wyzwalacz we frontend Zabbix.

Należy pamiętać, że oficjalne szablony Zabbix zawierają predefiniowane wyzwalacze.
Zobacz przewodnik dotyczący zastosowania szablonu do monitorowania systemu Linux.
Możliwe jest również masowe aktualizowanie skonfigurowanych wyzwalaczy.

Inne materiały:

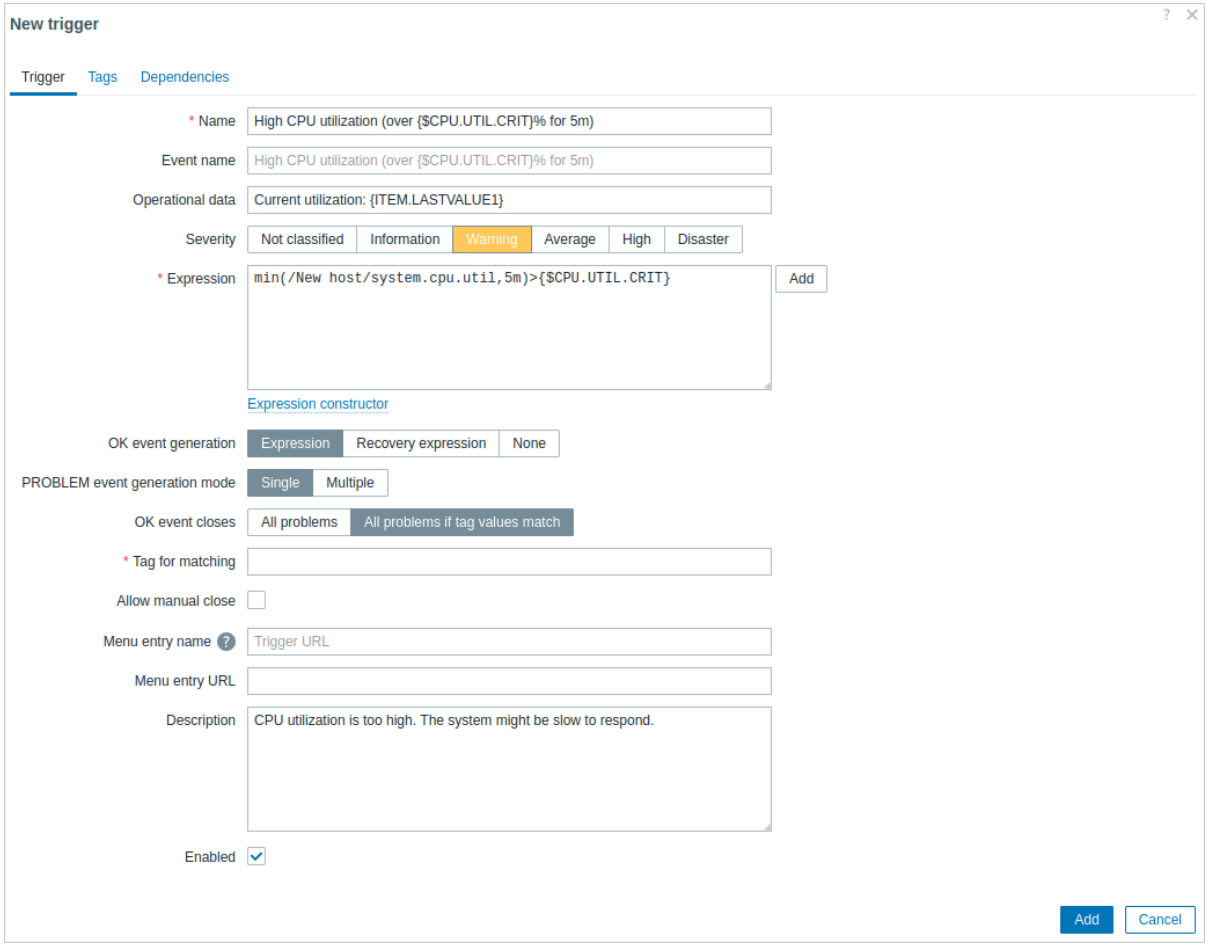
- Wykrywanie problemów za pomocą wyzwalaczy (ogólne wprowadzenie)
- Przykłady wyzwalaczy (według zastosowania)
- Wyrażenie wyzwalacza i funkcje (szczegóły składni)
- Najlepsze praktyki

Konfiguracja

Aby skonfigurować wyzwalacz, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Hosts*.
2. Kliknij *Triggers* w wierszu hosta.
3. Kliknij *Create trigger* po prawej stronie (lub nazwę wyzwalacza, aby edytować istniejący wyzwalacz).
4. Wprowadź parametry wyzwalacza w formularzu.

Karta **Trigger** zawiera wszystkie podstawowe atrybuty wyzwalacza.



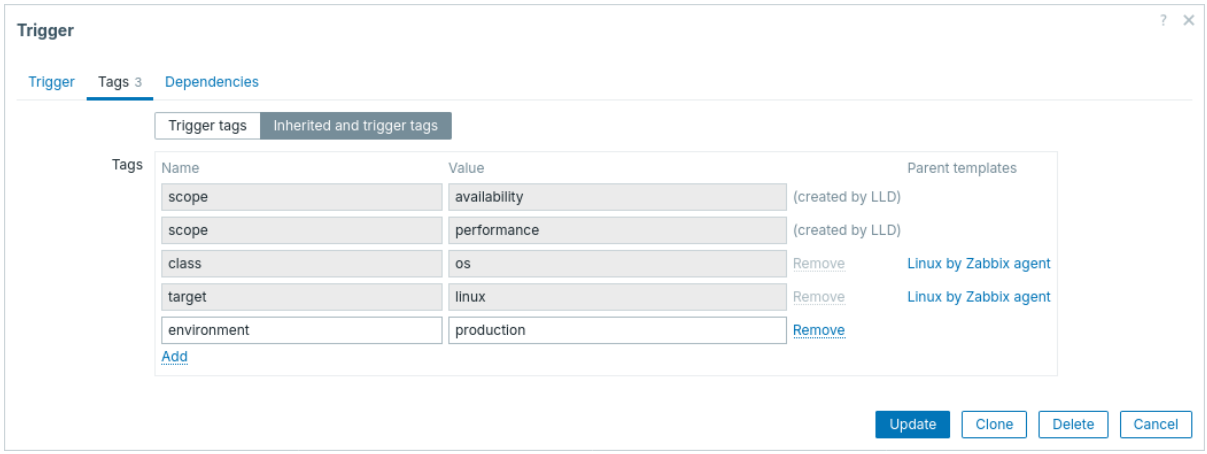
Wszystkie wymagane pola są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Nazwa wyzwalacza. Obsługiwane makra to: {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.*} oraz makra użytkownika {$MACRO}. Makra **$1, $2...$9** można użyć do odwołania się do pierwszej, drugiej... dziewiątej stałej wyrażenia. *Uwaga*: makra $1-$9 zostaną poprawnie rozwinięte, jeśli odnoszą się do stałych w stosunkowo prostych, nieskomplikowanych wyrażeniach. Na przykład nazwa ”Processor load above $1 on {HOST.NAME}” zostanie automatycznie zmieniona na ”Processor load above 5 on New host”, jeśli wyrażenie ma postać last(/New host/system.cpu.load[percpu,avg1])>5.
Event name	Jeśli zostanie zdefiniowana, ta nazwa zostanie użyta do utworzenia nazwy zdarzenia problemu. Domyślnie nazwa zdarzenia jest taka sama jak nazwa wyzwalacza. Nazwa zdarzenia może służyć do tworzenia czytelnych alertów zawierających dane problemu (zobacz przykład). Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, a dodatkowo makra wyrażenia {TIME}, {TIMESTAMP} oraz {?EXPRESSION}.

Parameter	Description
Operational data	Wprowadź ciąg znaków z makrami, aby wyświetlać dynamiczne dane w czasie rzeczywistym w *Monitoring* > *Problems*. Alternatywnie pozostaw to pole puste, aby wyświetlić najnowsze wartości wszystkich pozycji z wyrażenia wyzwalacza. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, z możliwością dynamicznego rozwijania. Na przykład: {ITEM.VALUE<1-9>} rozwija się do wartości pozycji w momencie zmiany stanu wyzwalacza (utworzenie problemu, rozwiązanie, ręczne zamknięcie lub zamknięcie przez korelację). {ITEM.LASTVALUE<1-9>} rozwija się do najnowszych wartości pozycji. Należy pamiętać, że ręczne zamknięcie problemu nie powoduje utworzenia nowej wartości, więc oba makra nadal będą pokazywać wartość z czasu wystąpienia problemu. Należy też pamiętać, że oba makra rozwijają się do *UNKNOWN*, jeśli najnowsza wartość jest starsza niż *Max history display period* (zobacz *Administration > General*).
Severity	Ustaw wymaganą severity wyzwalacza, klikając odpowiednie przyciski.
Expression	Logiczne wyrażenie używane do definiowania warunków problemu. Obsługiwane są sufiksy czasu oraz sufiksy rozmiaru pamięci. Problem zostaje utworzony po spełnieniu wszystkich warunków zawartych w wyrażeniu, tj. gdy wyrażenie przyjmuje wartość TRUE. Problem zostanie rozwiązany, gdy tylko wyrażenie przyjmie wartość FALSE, chyba że w *Recovery expression* określono dodatkowe warunki odzyskiwania.
OK event generation	Opcje generowania zdarzeń OK: **Expression** - zdarzenia OK są generowane na podstawie tego samego wyrażenia co zdarzenia problemu; **Recovery expression** - zdarzenia OK są generowane, jeśli wyrażenie problemu przyjmuje wartość FALSE, a wyrażenie odzyskiwania przyjmuje wartość TRUE; **None** - w tym przypadku wyzwalacz nigdy sam nie powróci do stanu OK.
Recovery expression	Logiczne wyrażenie (opcjonalne) definiujące dodatkowe warunki, które muszą zostać spełnione, zanim problem zostanie rozwiązany, po tym jak pierwotne wyrażenie problemu zostało już ocenione jako FALSE. Wyrażenie odzyskiwania jest przydatne w przypadku histerezy wyzwalacza. Nie jest **możliwe** rozwiązanie problemu wyłącznie przez wyrażenie odzyskiwania, jeśli wyrażenie problemu nadal ma wartość TRUE. To pole jest dostępne tylko wtedy, gdy dla *OK event generation* wybrano opcję 'Recovery expression'.
PROBLEM event generation mode	Tryb generowania zdarzeń problemu: **Single** - pojedyncze zdarzenie jest generowane, gdy wyzwalacz po raz pierwszy przechodzi w stan 'Problem'; **Multiple** - zdarzenie jest generowane przy *każdej* ocenie 'Problem' wyzwalacza.
OK event closes	Wybierz, czy zdarzenie OK zamyka: **All problems** - wszystkie problemy tego wyzwalacza; **All problems if tag values match** - tylko te problemy wyzwalacza, których wartości tagów zdarzeń są zgodne.
Tag for matching	Wprowadź nazwę tagu zdarzenia, która ma być używana do korelacji zdarzeń. To pole jest wyświetlane, jeśli dla właściwości *OK event closes* wybrano opcję 'All problems if tag values match' i w takim przypadku jest wymagane.
Allow manual close	Zaznacz, aby zezwolić na ręczne zamykanie zdarzeń problemu generowanych przez ten wyzwalacz. Ręczne zamknięcie jest możliwe podczas potwierdzania zdarzeń problemu.
Menu entry name	Jeśli pole nie jest puste, wpisana tutaj nazwa (do 64 znaków) jest używana w kilku miejscach frontend jako etykieta dla adresu URL wyzwalacza określonego w parametrze *Menu entry URL*. Jeśli pole jest puste, używana jest domyślna nazwa *Trigger URL*. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, a dodatkowo {EVENT.ID}, {HOST.ID} oraz {TRIGGER.ID}.
Menu entry URL	Jeśli pole nie jest puste, wpisany tutaj adres URL (do 2048 znaków) jest dostępny jako łącze w menu zdarzenia w kilku miejscach frontend, na przykład po kliknięciu nazwy problemu w widżecie pulpitu *Monitoring* > *Problems* lub *Problems*. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza, a dodatkowo {EVENT.ID}, {HOST.ID} oraz {TRIGGER.ID}. Uwaga: makra użytkownika z wartościami tajnymi nie zostaną rozwinięte w adresie URL.
Description	Pole tekstowe służące do podania dodatkowych informacji o tym wyzwalaczu. Może zawierać instrukcje dotyczące naprawy konkretnego problemu, dane kontaktowe odpowiedzialnego personelu itp. Obsługiwany jest ten sam zestaw makr co w nazwie wyzwalacza.

Parameter	Description
Enabled	Odznaczenie tego pola spowoduje wyłączenie wyzwalacza, jeśli jest to wymagane. Problemy wyłączonego wyzwalacza nie są już wyświetlane w frontend, ale nie są usuwane.

Karta **Tags** umożliwia zdefiniowanie tagów na poziomie wyzwalacza. Wszystkie problemy tego wyzwalacza będą oznaczone wartościami wprowadzonymi tutaj.



Dodatkowo opcja *Inherited and trigger tags* umożliwia wyświetlanie tagów zdefiniowanych na poziomie szablonu, jeśli wyzwalacz pochodzi z tego szablonu. Jeśli istnieje wiele szablonów z tym samym tagiem, tagi te są wyświetlane tylko raz, a nazwy szablonów są oddzielone przecinkami. Wyzwalacz nie ”dziedziczy” i nie wyświetla tagów na poziomie hosta.

Parameter	Description
Name/Value	Ustaw niestandardowe tagi do oznaczania zdarzeń wyzwalacza. Tagi są parą nazwy tagu i wartości. Możesz użyć tylko nazwy albo połączyć ją z wartością. Wyzwalacz może mieć kilka tagów o tej samej nazwie, ale różnych wartościach. Makra użytkownika, makra użytkownika z kontekstem, makra niskopoziomowego wykrywania oraz funkcje makr functions z `{{ITEM.VALUE}}`, `{{ITEM.LASTVALUE}}` są obsługiwane w tagach zdarzeń. Makra niskopoziomowego wykrywania mogą być używane wewnątrz kontekstu makra. Makra {TRIGGER.ID} są obsługiwane w wartościach tagów wyzwalacza. Może to być przydatne do identyfikowania wyzwalaczy utworzonych na podstawie prototypów wyzwalaczy i na przykład do tłumienia problemów z tych wyzwalaczy podczas konserwacji. Jeśli łączna długość rozwiniętej wartości przekroczy 255, zostanie ona obcięta do 255 znaków. Zobacz wszystkie makra obsługiwane dla tagów zdarzeń. Tagi zdarzeń mogą być używane do korelacji zdarzeń, w warunkach akcji, a także będą widoczne w *Monitoring* > *Problems* lub w widżecie *Problems*.

Karta **Dependencies** zawiera wszystkie zależności wyzwalacza.

Kliknij *Add*, aby dodać nową zależność.

Note:
Wyzwalacz można również skonfigurować, otwierając istniejący, klikając przycisk *Clone*, a następnie zapisując go pod inną nazwą.

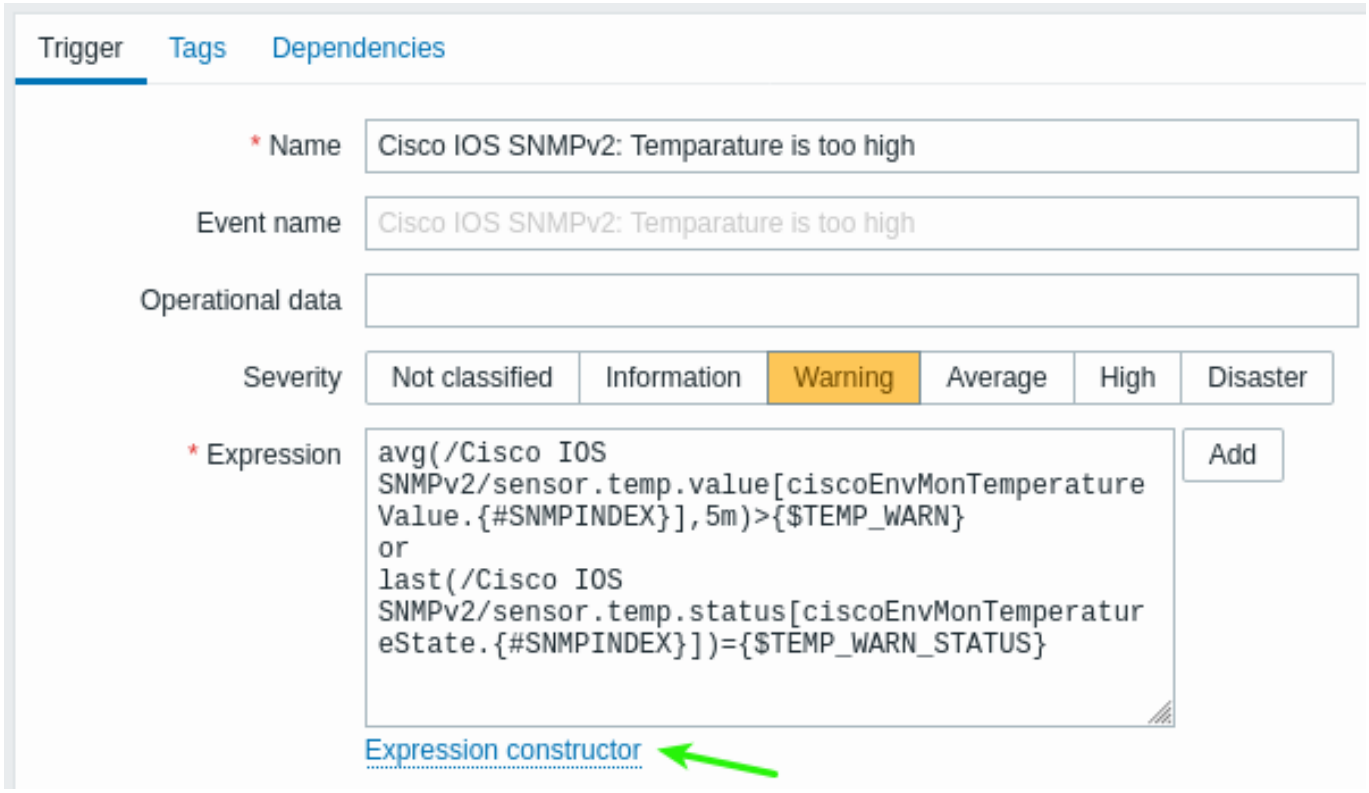
Testowanie wyrażenia wyzwalacza

Możliwe jest sprawdzenie, jaki byłby wynik wyrażenia w zależności od otrzymanej wartości.

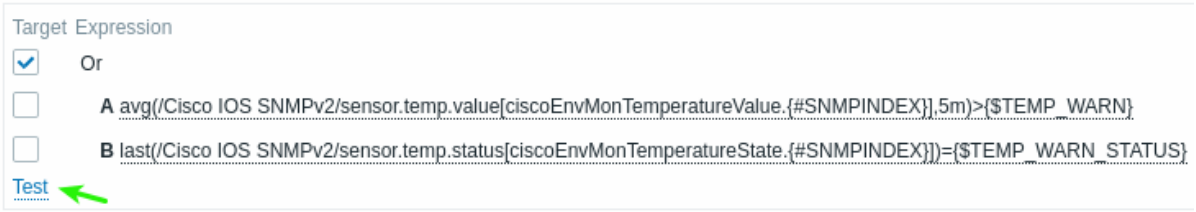
Jako przykład przyjęto następujące wyrażenie z oficjalnego szablonu:

```
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}
```

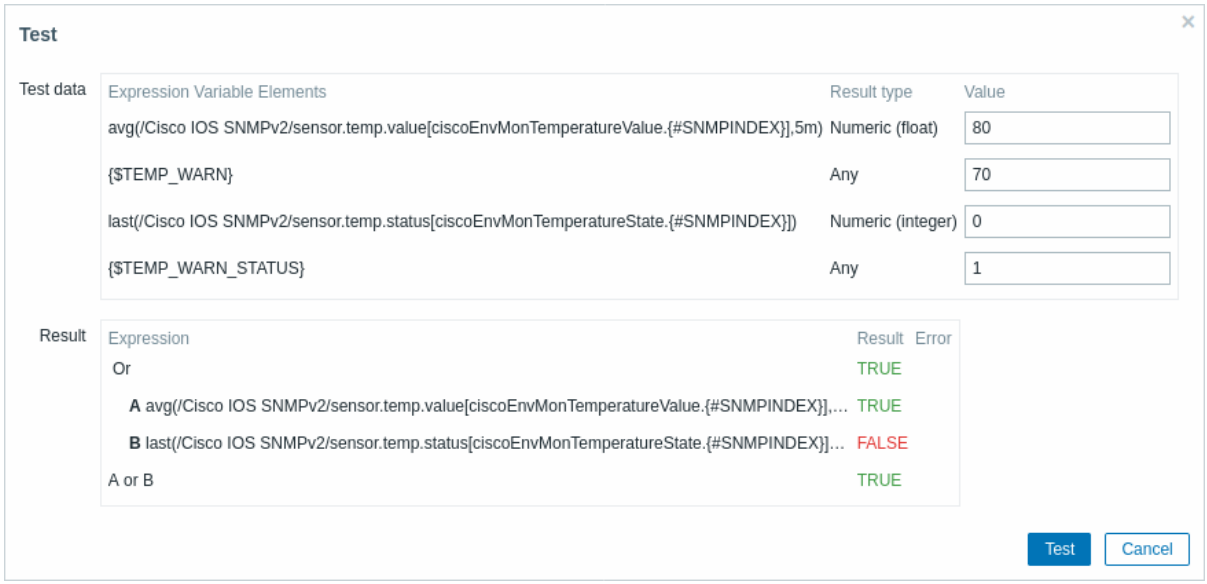
Aby przetestować wyrażenie, kliknij *Expression constructor* pod polem wyrażenia.



W Expression constructor wyświetlane są wszystkie pojedyncze wyrażenia. Aby otworzyć okno testowania, kliknij *Test* pod listą wyrażeń.



W oknie testowania możesz wprowadzić przykładowe wartości (’80’, ’70’, ’0’, ’1’ w tym przykładzie), a następnie sprawdzić wynik wyrażenia, klikając przycisk *Test*.



Można zobaczyć wynik poszczególnych wyrażeń, a także całego wyrażenia.

”TRUE” oznacza, że określone wyrażenie jest poprawne. W tym konkretnym przypadku A wartość ”80” jest większa niż określona wartość `{$TEMP_WARN}`, czyli ”70” w tym przykładzie. Zgodnie z oczekiwaniami pojawia się wynik ”TRUE”.

”FALSE” oznacza, że określone wyrażenie jest niepoprawne. W tym konkretnym przypadku B `{$TEMP_WARN_STATUS}` ”1” musi być równe określonej wartości, czyli ”0” w tym przykładzie. Zgodnie z oczekiwaniami pojawia się wynik ”FALSE”.

Wybrany typ wyrażenia to ”OR”. Jeśli co najmniej jeden z określonych warunków (w tym przypadku A lub B) ma wartość TRUE, wynik końcowy również będzie TRUE. Oznacza to, że bieżąca wartość przekracza wartość ostrzegawczą i wystąpił problem.

Wyrażenie wyzwalacza i funkcje

Przegląd

Ta strona opisuje składnię wyrażeń wyzwalaczy oraz szczegóły ich obliczania.

Składnia prostego wyrażenia jest następująca:

`function(/host/item,time_period)<operator><constant>`

W tym wyrażeniu pierwszy operand (po lewej stronie operatora) to funkcja z jej parametrami w nawiasach (zwykle pozycja danych i okres czasu).

Funkcja służy do analizowania odebranych danych w ustawionym okresie czasu, czego wynikiem jest obliczona wartość.

Ta wartość jest następnie porównywana z drugim operandem przy użyciu operatora. W tym przykładzie drugi operand jest stałą, ale może to być również inna funkcja.

Na przykład:

`min(/Zabbix server/net.if.in[eth0,bytes],5m)>100K`

Ten wyzwalacz uruchamia się, jeśli liczba odebranych bajtów na `eth0` w ciągu ostatnich pięciu minut była cały czas większa niż 100 kilobajtów. W takim przypadku wyrażenie ma wartość prawda i tworzony jest problem.

Wyrażenia wyzwalaczy są niezwykle elastyczne. W bardziej złożonych wyrażeniach można łączyć wiele funkcji, operatorów i stałych.

Zobacz także:

- Przykłady wyzwalaczy
- Wykrywanie problemów za pomocą wyzwalaczy (ogólne wprowadzenie)

Funkcje

Funkcje umożliwiają analizę zebranych wartości, na przykład obliczanie średniej lub wyszukiwanie określonego ciągu znaków.

Kliknij odpowiednią grupę funkcji, aby wyświetlić więcej szczegółów.

Grupa funkcji	Funkcje
Funkcje agregujące	avg, bucket_percentile, count, histogram_quantile, item_count, kurtosis, mad, max, min, skewness, stddevpop, stddevsamp, sum, sumofsquares, varpop, varsamp
Funkcje Foreach	avg_foreach, bucket_rate_foreach, count_foreach, exists_foreach, last_foreach, max_foreach, min_foreach, sum_foreach
Funkcje bitowe	bitand, bitlshift, bitnot, bitor, bitrshift, bitxor
Funkcje daty i czasu	date, dayofmonth, dayofweek, now, time
Funkcje historii	change, changecount, count, countunique, find, first, firstclock, fuzzytime, jsonpath, last, lastclock, logeventid, logseverity, logsource, logtimestamp, monodec, monoinc, nodata, percentile, rate, xmlxpath
Funkcje trendów	baselinedev, baselinewma, trendavg, trendcount, trendmax, trendmin, trendstl, trendsum
Funkcje matematyczne	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
Funkcje operatorów	between, in

Grupa funkcji	Funkcje
Funkcje predykcyjne	forecast, timeleft
Funkcje ciągów znaków	ascii, bitlength, bytelength, char, concat, contains, insert, jsonpath, left, length, ltrim, mid, repeat, replace, right, rtrim, substring, trim, xmlxpath

O ile nie określono inaczej, funkcje te są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje Foreach są obsługiwane tylko w przypadku obliczeń agregujących.

Zazwyczaj funkcje zwracają wartości liczbowe do porównania. W przypadku zwracania ciągów znaków porównanie jest możliwe za pomocą operatorów **=** i **<>** (zobacz przykład Wykrywanie niepasującego oprogramowania na różnych hostach).

Parametry funkcji

Parametry funkcji umożliwiają określenie:

- klucza pozycji (w postaci `/host/key`) dla funkcji odwołujących się do historii pozycji hosta
- okresu (oraz innych parametrów specyficznych dla funkcji)
- innych wyrażeń

Klucz pozycji

Pozycja, do której następuje odwołanie, musi być w obsługiwanym stanie (z wyjątkiem funkcji **nodata()**, która jest obliczana również dla nieobsługiwanych pozycji).

Pominięcie nazwy hosta w pierwszym parametrze (tj. jak w `function(//key,parameter,...)`) jest obsługiwane tylko w określonych kontekstach:

- W formule pozycji obliczanych
- W makrach wyrażeń, które mogą być używane w:
  - polu *Nazwa zdarzenia*
  - nazwie wykresu
  - etykiecie elementów mapy „Host” i „Trigger” map elements

W tych kontekstach można również użyć makra `{HOST.HOST}`. Makro `{HOST.HOST<1-9>}` może być użyte w przypadku pola *Nazwa zdarzenia* oraz elementu mapy „Trigger”, aby odwołać się do określonej pozycji w wyrażeniu wyzwalacza. Gdy nazwa hosta jest pominięta lub zastąpiona przez `{HOST.HOST}` w tych kontekstach, odwołanie wskazuje na pierwszą pozycję w wyrażeniu wyzwalacza lub na pierwszą pozycję na wykresie. Poza tymi obsługiwanymi kontekstami pominięcie nazwy hosta w wyrażeniach wyzwalaczy spowoduje błąd. Zobacz przykład Compare long-term CPU loads, aby zobaczyć ilustrację użycia podwójnego ukośnika w makrach Nazwy zdarzenia.

Okres

Parametry specyficzne dla funkcji umieszcza się za kluczem pozycji i oddziela od niego przecinkiem.

Większość funkcji numerycznych przyjmuje **okres** jako parametr. Umożliwia on określenie interesującego nas przedziału. Można go określić jako okres (30s, 10m, 1h) lub zakres wartości (#5 - pięć najnowszych wartości).

Do wskazania okresu można użyć sekund lub przyrostków czasu. Jeśli parametr jest poprzedzony znakiem hash, ma inne znaczenie:

Expression	Description
sum(/host/key,**10m)**	Suma wartości z ostatnich 10 minut.
sum(/host/key,**#10)**	Suma dziesięciu ostatnich wartości.

Parametry ze znakiem hash mają inne znaczenie w przypadku funkcji **last** - oznaczają N-tą poprzednią wartość, więc dla wartości 30, 70, 20, 60, 50 (od najnowszej do najstarszej):

- `last(/host/key,#2)` zwróci „70”
- `last(/host/key,#5)` zwróci „50”

Okres jest mierzony do chwili „teraz” - gdzie „teraz” oznacza czas ostatniego przeliczenia wyzwalacza (zobacz Częstotliwość obliczeń); „teraz” nie oznacza bieżącego czasu serwera.

Okres określa:

- Wszystkie wartości między „teraz - okres” a „teraz” (lub, po zastosowaniu przesunięcia czasowego, między „teraz - przesunięcie czasowe - okres” a „teraz - przesunięcie czasowe”).
- Określoną liczbę wartości z przeszłości, do chwili „teraz”.

Jeśli w określonym okresie oceny nie istnieją żadne dane, wyzwalacz lub obliczana pozycja używająca tej funkcji staje się nieobsługiwana.

Należy pamiętać, że:

- Jeśli w wyzwalaczu używana jest tylko jedna funkcja (odwołująca się do historii danych), „teraz” zawsze oznacza czas otrzymania najnowszej wartości. Na przykład jeśli ostatnia wartość została otrzymana godzinę temu, okres będzie liczony do najnowszej wartości sprzed godziny.
- Nowy wyzwalacz jest obliczany natychmiast po otrzymaniu pierwszej wartości (funkcje historii); w przypadku funkcji daty i czasu oraz nodata() zostanie obliczony w ciągu 30 sekund. Oznacza to, że wyzwalacz zostanie obliczony, nawet jeśli od jego utworzenia nie upłynął jeszcze określony okres (na przykład godzina). Wyzwalacz zostanie również obliczony po otrzymaniu pierwszej wartości, nawet jeśli zakres czasu został ustawiony na przykład na dziesięć najnowszych wartości.

Przesunięcie czasowe

Opcjonalne przesunięcie czasowe jest obsługiwane w przypadku czasu lub liczby wartości jako parametru funkcji. Ten parametr umożliwia odwoływanie się do danych z okresu w przeszłości.

Przesunięcie czasowe rozpoczyna się od `now` - określającego bieżący czas, po którym następuje `+N<jednostka czasu>` lub `-N<jednostka czasu>` - w celu dodania lub odjęcia N jednostek czasu.

Na przykład `avg(/host/key,1h:now-1d)` zwróci średnią wartość z godziny sprzed jednego dnia.

Attention:
Przesunięcie czasowe określone w miesiącach (M) i latach (y) jest obsługiwane tylko przez funkcje trendów. Inne funkcje obsługują sekundy (s), minuty (m), godziny (h), dni (d) i tygodnie (w).

Przesunięcie czasowe z bezwzględnymi okresami

W parametrze przesunięcia czasowego obsługiwane są bezwzględne okresy, na przykład od północy do północy dla dnia, od poniedziałku do niedzieli dla tygodnia, od pierwszego do ostatniego dnia miesiąca dla miesiąca.

Przesunięcie czasowe dla bezwzględnych okresów rozpoczyna się od `now` - określającego bieżący czas, po którym może następować dowolna liczba operacji czasu: `/<jednostka czasu>` - definiuje początek i koniec jednostki czasu, na przykład od północy do północy dla dnia, `+N<jednostka czasu>` lub `-N<jednostka czasu>` - w celu dodania lub odjęcia N jednostek czasu.

Należy pamiętać, że wartość przesunięcia czasowego może być większa lub równa 0, natomiast minimalna wartość okresu wynosi 1.

Parametr	Opis
1d:now/d	Wczoraj
1d:now/d+1d	Dzisiaj
2d:now/d+1d	Ostatnie 2 dni
1w:now/w	Ostatni tydzień
1w:now/w+1w	Ten tydzień

Inne wyrażenia

Parametry funkcji mogą zawierać inne wyrażenia, jak w poniższej składni:

```
min(min(/host/key,1h),min(/host2/key2,1h)*10)
```

Należy pamiętać, że inne wyrażenia nie mogą być używane, jeśli funkcja odwołuje się do historii pozycji. Na przykład poniższa składnia **nie** jest dozwolona:

~~`min(/host/key,#5*10)`~~

Chociaż inne wyrażenia wyzwalaczy jako parametry funkcji są w wyzwalaczach ograniczone do funkcji nieodwołujących się do historii, to ograniczenie nie ma zastosowania w pozycjach obliczanych.

Operatory

Następujące operatory są obsługiwane dla wyzwalaczy **(w malejącej kolejności priorytetu wykonywania)**:

Priorytet	Operator	Definicja	**Uwagi dotyczące wartości nieznanych**	Wymuś rzutowanie operandu na float [1]
1	**-**	Jednoargumentowy minus	**-**Unknown → Unknown	Tak
2	**not**	Logiczne NOT	**not** Unknown → Unknown	Tak
3	*	Mnożenie	0 * Unknown → Unknown (tak, Unknown, a nie 0 - aby nie utracić Unknown w operacjach arytmetycznych) 1.2 * Unknown → Unknown	Tak
	/	Dzielenie	Unknown / 0 → error Unknown / 1.2 → Unknown 0.0 / Unknown → Unknown	Tak
4	+	Dodawanie arytmetyczne	1.2 + Unknown → Unknown	Tak
	-	Odejmowanie arytmetyczne	1.2 - Unknown → Unknown	Tak
5	<	Mniejsze niż. Operator jest zdefiniowany jako: A<B ⇔ (A<B-0.000001)	1.2 < Unknown → Unknown	Tak
	<=	Mniejsze niż lub równe. Operator jest zdefiniowany jako: A<=B ⇔ (A≤B+0.000001)	Unknown <= Unknown → Unknown	Tak
	>	Większe niż. Operator jest zdefiniowany jako: A>B ⇔ (A>B+0.000001)		Tak
	>=	Większe niż lub równe. Operator jest zdefiniowany jako: A>=B ⇔ (A≥B-0.000001)		Tak
6	=	Równe. Operator jest zdefiniowany jako: A=B ⇔ (A≥B-0.000001) and (A≤B+0.000001)		Nie [1]

Priorytet	Operator	Definicja	**Uwagi dotyczące wartości nieznanych**	Wymuś rzutowanie operandu na float [1]
	<>	Różne. Operator jest zdefiniowany jako: A<>B ⇔ (A<B-0.000001) or (A>B+0.000001)		Nie [1]
7	**and**	Logiczne AND	0 **and** Unknown → 0 1 **and** Unknown → Unknown Unknown **and** Unknown → Unknown	Tak
8	**or**	Logiczne OR	1 **or** Unknown → 1 0 **or** Unknown → Unknown Unknown **or** Unknown → Unknown	Tak

[1] Operand typu string jest nadal rzutowany na wartość numeryczną, jeśli:

- drugi operand jest numeryczny
- na operandzie użyto operatora innego niż **=** lub **<>**

(Jeśli rzutowanie się nie powiedzie - operand numeryczny jest rzutowany na operand typu string i oba operandy są porównywane jako stringi.)

Operatory **not**, **and** i **or** rozróżniają wielkość liter i muszą być zapisane małymi literami. Muszą też być otoczone spacjami lub nawiasami.

Wszystkie operatory, z wyjątkiem jednoargumentowego **-** i **not**, mają lewostronną łączność. Jednoargumentowe **-** i **not** nie są łączne (co oznacza, że należy używać **-(-1)** i **not (not 1)** zamiast **--1** i **not not 1**).

Wynik obliczenia:

- operatory **<**, **<=**, **>**, **>=**, **=**, **<>** zwracają '1' w wyrażeniu wyzwalacza, jeśli określona relacja jest prawdziwa, oraz '0', jeśli jest fałszywa. Jeśli co najmniej jeden operand ma wartość Unknown, wynikiem jest Unknown;
- **and** dla znanych operandów zwraca '1', jeśli oba jego operandy są różne od '0'; w przeciwnym razie zwraca '0'; dla nieznanych operandów **and** zwraca '0' tylko wtedy, gdy jeden operand jest równy '0'; w przeciwnym razie zwraca 'Unknown';
- **or** dla znanych operandów zwraca '1', jeśli którykolwiek z jego operandów jest różny od '0'; w przeciwnym razie zwraca '0'; dla nieznanych operandów **or** zwraca '1' tylko wtedy, gdy jeden operand jest różny od '0'; w przeciwnym razie zwraca 'Unknown';
- wynik operatora negacji logicznej **not** dla znanego operandu wynosi '0', jeśli wartość jego operandu jest różna od '0'; '1', jeśli wartość jego operandu jest równa '0'. Dla nieznanego operandu **not** zwraca 'Unknown'.

Nieznany stan wyrażenia

Możliwe jest, że w wyrażeniu wyzwalacza pojawi się nieznany operand, gdy:

- używana jest nieobsługiwana pozycja
- ocena funkcji dla obsługiwanej pozycji kończy się błędem

W takim przypadku wyrażenie wyzwalacza zazwyczaj jest oceniane jako `Unknown` (ponieważ nie można go ocenić)

Możliwe jest otrzymywanie powiadomień o nieznanych wyzwalaczach.

Wyjątki

Pomimo nieznanego operandu, wyrażenia wyzwalacza mogą w niektórych przypadkach zostać ocenione do znanego wyniku (Problem/OK):

- Funkcja `nodata()` jest oceniana niezależnie od tego, czy odwoływana pozycja jest obsługiwana, czy nie.
- Wyrażenia z AND/OR mogą zostać ocenione do znanego wyniku w dwóch przypadkach:
  - **Przypadek 1**: "`1 or some_function(unsupported_item1) or some_function(unsupported_item2) or ...`" jest oceniane do znanego wyniku ('1' lub "Problem"),
  - **Przypadek 2**: "`0 and some_function(unsupported_item1) and some_function(unsupported_item2) and ...`" jest oceniane do znanego wyniku ('0' lub "OK").

- Jeśli ocena funkcji dla obsługiwanej pozycji kończy się błędem, wartość funkcji staje się `Unknown` i bierze ona udział jako nieznany operand w dalszej ocenie wyrażenia.

Nieznane operandy mogą ”zniknąć” tylko w wyrażeniach logicznych, jak opisano powyżej. W wyrażeniach arytmetycznych nieznane operandy zawsze prowadzą do `Unknown` (z wyjątkiem dzielenia przez 0).

Attention:
Nieznany stan wyrażenia nie zmienia stanu wyzwalacza (Problem/OK). Dlatego jeśli stan wyzwalacza był ”Problem” (zobacz Przypadek 1), pozostaje on w stanie problemu nawet wtedy, gdy znana część zostanie rozwiązana (’1’ staje się ’0’), ponieważ wyrażenie jest teraz oceniane jako `Unknown`, a to nie zmienia stanu wyzwalacza.

Jeśli wyrażenie wyzwalacza z kilkoma nieobsługiwanymi pozycjami zostanie ocenione jako `Unknown`, komunikat błędu w frontend odnosi się do ostatniej ocenionej nieobsługiwanej pozycji.

Buforowanie wartości

Wartości wymagane do obliczania wyzwalaczy są buforowane przez serwer Zabbix. Z tego powodu obliczanie wyzwalaczy przez pewien czas po ponownym uruchomieniu serwera powoduje większe obciążenie bazy danych.

Bufor wartości nie jest czyszczony po usunięciu wartości historii pozycja (ręcznie lub przez housekeeper), dlatego serwer będzie używać wartości z pamięci podręcznej, dopóki nie będą one starsze niż okresy czasu zdefiniowane w funkcjach wyzwalaczy lub dopóki serwer nie zostanie ponownie uruchomiony.

Note:
Jeśli w pamięci podręcznej nie ma aktualnych danych i w funkcji nie zdefiniowano okresu odpytywania, Zabbix domyślnie cofnie się maksymalnie o tydzień, aby odpytać bazę danych o wartości historyczne.

Funkcje agregujące

O ile nie zaznaczono inaczej, wszystkie funkcje wymienione tutaj są obsługiwane w:

- wyrażeniach wyzwalaczy
- formułach obliczanych pozycji
- makrach wyrażeń

Funkcje agregujące mogą działać z:

- historią pozycji, na przykład `min(/host/key,1h)`
- funkcjami foreach jako jedynym parametrem, na przykład `min(last_foreach(/*/key))` (tylko w formułach obliczanych pozycji; aby użyć ich w wyzwalaczach, odwołaj się do obliczanej pozycji za pomocą jej klucza)

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
avg	Średnia wartość pozycji w zdefiniowanym okresie oceny.
bucket_percentile	Oblicza percentyl na podstawie kubełków histogramu.
count	Liczba wartości w tablicy zwróconej przez funkcję foreach.
histogram_quantile	Oblicza kwantyl φ na podstawie kubełków histogramu.
item_count	Liczba istniejących pozycji w konfiguracji, które pasują do kryteriów filtra.
kurtosis	”Szczytowość” rozkładu prawdopodobieństwa w zebranych wartościach w zdefiniowanym okresie oceny.
mad	Mediana odchyleń bezwzględnych w zebranych wartościach w zdefiniowanym okresie oceny.
max	Najwyższa wartość pozycji w zdefiniowanym okresie oceny.
min	Najniższa wartość pozycji w zdefiniowanym okresie oceny.
skewness	Asymetria rozkładu prawdopodobieństwa w zebranych wartościach w zdefiniowanym okresie oceny.
stddevpop	Odchylenie standardowe populacji w zebranych wartościach w zdefiniowanym okresie oceny.
stddevsamp	Odchylenie standardowe próby w zebranych wartościach w zdefiniowanym okresie oceny.
sum	Suma zebranych wartości w zdefiniowanym okresie oceny.
sumofsquares	Suma kwadratów w zebranych wartościach w zdefiniowanym okresie oceny.
varpop	Wariancja populacji zebranych wartości w zdefiniowanym okresie oceny.
varsamp	Wariancja próby zebranych wartości w zdefiniowanym okresie oceny.

Wspólne parametry

- `/host/key` jest wspólnym wymaganym pierwszym parametrem funkcji odwołujących się do historii pozycji host.
- `(sec|#num)<:time shift>` jest wspólnym drugim parametrem funkcji odwołujących się do historii pozycji host, gdzie:
  - **sec** - maksymalny okres oceny w sekundach (można używać przyrostków czasu).
  - **#num** - maksymalny zakres oceny w najnowszych zebranych wartościach (jeśli poprzedzony znakiem hash).
  - **time shift** (opcjonalny) umożliwia przesunięcie punktu oceny wstecz w czasie. Więcej informacji na temat określania przesunięcia w czasie zawiera sekcja więcej szczegółów.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `(sec|#num)<:time shift>` nigdy nie mogą być ujmowane w cudzysłów

avg(/host/key,(sec|#num)<:time shift>)

Średnia wartość pozycji w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przesunięcie czasu jest przydatne, gdy istnieje potrzeba porównania bieżącej wartości średniej ze średnią wartością sprzed pewnego czasu.

Przykłady:

```
avg(/host/key,1h) #oblicz średnią wartość za ostatnią godzinę do teraz
avg(/host/key,#5) #oblicz średnią wartość z pięciu najnowszych wartości
avg(/host/key,1h:now-1d) #oblicz średnią wartość dla jednej godziny od 25 godzin temu do 24 godzin temu od
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"])) #oblicz średnie obciążenie procesora
avg(/host/proc.num,5m)>300 #wyzwalacz, jeśli średnia liczba procesów w ciągu ostatnich 5 minut była powyże
```

bucket_percentile(item filter,time period,percentage)

Oblicza percentyl z bucketów histogramu.

Parametry:

- **item filter** - zobacz item filter;
- **time period** - zobacz time period;
- **percentage** - wartość procentowa (0-100).

Uwagi:

- Obsługiwane tylko w pozycjach obliczanych;
- Ta funkcja jest aliasem dla `histogram_quantile(percentage/100, bucket_rate_foreach(item filter, time period, 1))`.

Przykłady:

```
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>)

Liczba wartości w tablicy zwróconej przez funkcję foreach.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry:

- **func_foreach** - funkcja foreach, dla której należy policzyć liczbę zwróconych wartości. Szczegóły znajdują się w sekcji funkcje foreach. Należy pamiętać, że count_foreach i bucket_rate_foreach obsługują dodatkowe parametry.
- **item filter** - zobacz item filter;
- **time period** - zobacz time period;
- **operator** (musi być ujęty w podwójne cudzysłowy). Obsługiwane `operators`:
eq - równe
ne - nierówne
gt - większe
ge - większe lub równe
lt - mniejsze
le - mniejsze lub równe
like - dopasowuje, jeśli zawiera wzorzec (uwzględnia wielkość liter)
bitand - bitowe AND
regexp - dopasowanie wyrażenia regularnego podanego w `pattern` z uwzględnieniem wielkości liter
iregexp - dopasowanie wyrażenia regularnego podanego w `pattern` bez uwzględniania wielkości liter
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójne cudzysłowy); obsługiwany, jeśli *operator* jest określony w trzecim parametrze.

Uwagi:

- Użycie **count()** z funkcją foreach opartą na historii (max_foreach, avg_foreach itp.) może mieć wpływ na wydajność, natomiast użycie **exists_foreach()**, która działa wyłącznie na danych konfiguracyjnych, nie będzie miało takiego efektu.
- Opcjonalne parametry *operator* lub *pattern* nie mogą pozostać puste po przecinku, muszą zostać całkowicie pominięte.
- Przy *bitand* jako trzecim parametrze czwarty parametr `pattern` może być określony jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. count() oblicza "bitwise AND" z wartości i *maski*, a następnie porównuje wynik z *number_to_compare_with*. Jeśli wynik "bitwise AND" jest równy *number_to_compare_with*, wartość jest liczona.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *maskę* (bez '/').
- Przy *regexp* lub *iregexp* jako trzecim parametrze czwarty parametr `pattern` może być zwykłym wyrażeniem regularnym lub globalnym (zaczynającym się od '@') wyrażeniem regularnym. W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego. Na potrzeby dopasowania regexp wartości zmiennoprzecinkowe będą zawsze reprezentowane z 4 cyframi po '.'. Należy też pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę po przecinku.

Przykłady:

```
count(max_foreach(/*/net.if.in[*],1h)) #oblicz liczbę pozycji net.if.in, które otrzymały dane w ciągu osta
count(last_foreach(/*/vfs.fs.size[*,pused]),"gt",95) #oblicz liczbę systemów plików, w których wykorzystan
```

histogram_quantile(quantile,bucket1,value1,bucket2,value2,...)

Oblicza φ-kwantyl na podstawie kubełków histogramu.
 Obsługiwana funkcja foreach: *bucket_rate_foreach*.

Parametry:

- **quantile** — 0 ≤ φ ≤ 1;
- **bucketN, valueN** — ręcznie wprowadzone pary (>=2) parametrów lub odpowiedź bucket_rate_foreach.

Uwagi:

- Obsługiwane tylko w pozycjach obliczanych;
- Funkcjonalnie odpowiada funkcji 'histogram_quantile' w PromQL;
- Zwraca -1, jeśli wartości ostatniego kubełka 'Infinity' ("+*inf*") są równe 0.

Przykłady:

```
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)

Liczba istniejących pozycji w konfiguracji, które odpowiadają kryteriom filtra.
 Obsługiwany typ wartości: *Integer*.

Parametr:

- **item filter** - kryteria wyboru pozycji, umożliwia odwoływanie się według grupy hostów, hosta, klucza pozycji oraz tagów. Obsługiwane są symbole wieloznaczne. Więcej informacji można znaleźć w item filter.

Uwagi:

- Obsługiwane tylko w pozycjach obliczanych;
- Działa jako alias funkcji *count(exists_foreach(item_filter))*.

Przykłady:

```
item_count(/*/agent.ping?[group="Host group 1"]) #oblicza liczbę hostów z pozycją agent.ping w "Host group
```

kurtosis(/host/key,(sec|#num)<:time shift>)

„Spiczastość" rozkładu prawdopodobieństwa wartości zebranych w zdefiniowanym okresie obliczania. Zobacz także: Kurtoza.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
kurtosis(/host/system.cpu.util[,user],5m)>5 #wyzwalacz przy gwałtownych skokach wykorzystania CPU w ciągu
```

mad(/host/key,(sec|#num)<:time shift>)

Mediana odchyleń bezwzględnych w zebranych wartościach w zdefiniowanym okresie obliczeniowym. Zobacz także: Median absolute deviation.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
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>)

Najwyższa wartość pozycji w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
max(/host/key,1h) - min(/host/key,1h) #oblicza różnicę między wartościami maksymalną i minimalną w ciągu c
max(last_foreach(/*/vfs.fs.size[*,pused]?[group="Linux servers"])) #zwraca najwyższe użycie dysku na wszys
```

min(/host/key,(sek|#num)<:przesunięcie czasu>)

Najniższa wartość pozycji w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
max(/host/key,1h) - min(/host/key,1h) #oblicza różnicę między wartościami maksymalną i minimalną w ciągu c
min(last_foreach(/*/vfs.file.contents["/sys/class/net/enp0s3/operstate"]?[group="Linux servers"])) #zwraca
```

skewness(/host/key,(sec|#num)<:time shift>)

Asymetria rozkładu prawdopodobieństwa zebranych wartości w zdefiniowanym okresie oceny. Zobacz także: Skewness.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
skewness(/host/vm.memory.size[used],5m)<-20 #trigger on sudden negative drops indicating crashes/restarts
```

stddevpop(/host/key,(sec|#num)<:time shift>)

Odchylenie standardowe populacji w zebranych wartościach w zdefiniowanym okresie obliczania. Zobacz także: Odchylenie standardowe.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
stddevpop(/host/system.cpu.util[,user],10m)>7 #wyzwalacz, jeśli użycie CPU silnie się waha (odchylenie sta
```

stddevsamp(/host/key,(sec|#num)<:time shift>)

Odchylenie standardowe z próby dla zebranych wartości w zdefiniowanym okresie obliczeniowym. Zobacz także: Odchylenie standardowe.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Do działania tej funkcji wymagane są co najmniej dwie wartości danych.

Przykłady:

```
stddevsamp(/host/system.cpu.util[,user],10m)>7 #wyzwalacz, jeśli użycie CPU silnie się waha (odchylenie st
```

sum(/host/key,(sec|#num)<:time shift>)

Suma zebranych wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwane funkcje foreach: *avg_foreach*, *count_foreach*, *exists_foreach*, *last_foreach*, *max_foreach*, *min_foreach*, *sum_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
sum(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #oblicza całkowity przychodzący ruch sieciowy d
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"])) #oblicza całkowitą przestrzeń dyskową d
sum(last_foreach(/*/net.dns[*,*,*])) #oblicza całkowitą liczbę pomyślnych kontroli DNS
```

sumofsquares(/host/key,(sec|#num)<:time shift>)

Suma kwadratów zebranych wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
sumofsquares(/host/calculated.net.if.rate,1m) #suma kwadratów dla szybkości ruchu sieciowego (przychodzące
sumofsquares(last_foreach(/*/net.if.in[*]?[group="Linux servers"])) #suma kwadratów najnowszych wartości p
```

varpop(/host/key,(sec|#num)<:time shift>)

Wariancja populacji zebranych wartości w zdefiniowanym okresie obliczania. Zobacz także: Wariancja.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Przykłady:

```
varpop(/host/system.cpu.util[,user],10m)>50 #wyzwalacz, jeśli użycie CPU silnie się waha (wariancja jest p
```

varsamp(/host/key,(sec|#num)<:time shift>)

Wariancja próbki zebranych wartości w zdefiniowanym okresie obliczania. Zobacz także: Wariancja.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Obsługiwana funkcja foreach: *last_foreach*.

Parametry: zobacz wspólne parametry.

Do działania tej funkcji wymagane są co najmniej dwie wartości danych.

Przykłady:

```
varsamp(/host/system.cpu.util[,user],10m)>50 #wyzwalacz, jeśli użycie CPU silnie się waha (wariancja próbk
```

Zobacz wszystkie obsługiwane funkcje.

Funkcje foreach

Przegląd

Funkcje foreach są używane w obliczeniach agregujących do zwracania jednej wartości agregowanej dla każdej pozycji wybranej przez użyty **filtr pozycji**. Zwracana jest tablica wartości.

Na przykład funkcja `avg_foreach` zwróci tablicę wartości, gdzie każda wartość jest *średnią* wartością historyczną wybranej pozycji w określonym przedziale czasu.

Attention:
Funkcje foreach są obsługiwane tylko w formułach pozycji obliczanej. Nie można ich wywoływać w zwykłych kluczach pozycji ani używać bezpośrednio do tworzenia wyrażeń wyzwalacza. W wyrażeniach wyzwalacza można użyć pozycji obliczanej, odwołując się do jej klucza.

Filtr pozycji jest częścią składni używanej przez funkcje foreach. W filtrze pozycji obsługiwane są symbole wieloznaczne, dzięki czemu wymagane pozycje można wybierać bardzo elastycznie.

Obsługiwane funkcje

Function	Description
avg_foreach	Zwraca średnią wartość dla każdej pozycji.
bucket_rate_foreach	Zwraca pary (górna granica kubełka, wartość szybkości) odpowiednie do użycia w funkcji `histogram_quantile()`, gdzie "górna granica kubełka" jest wartością parametru klucza pozycji zdefiniowaną przez <parameter number> parameter.
count_foreach	Zwraca liczbę wartości dla każdej pozycji.
exists_foreach	Zwraca '1' dla każdej włączonej pozycji.
last_foreach	Zwraca ostatnią wartość dla każdej pozycji.
max_foreach	Zwraca maksymalną wartość dla każdej pozycji.
min_foreach	Zwraca minimalną wartość dla każdej pozycji.
sum_foreach	Zwraca sumę wartości dla każdej pozycji.

Składnia funkcji

Funkcje foreach obsługują dwa wspólne parametry: `filtr pozycji` (szczegóły poniżej) oraz `okres czasu`:

```
foreach_function(item filter,time period)
```

Na przykład:

```
avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m)
```

zwróci pięciominutową średnią dla każdej pozycji „mysql.qps" w grupie serwerów MySQL.

Należy pamiętać, że niektóre funkcje obsługują dodatkowe parametry.

Składnia filtra pozycji

Filtr pozycji:

```
/host/key[parameters]?[conditions]
```

składa się z czterech części, gdzie:

- host - nazwa hosta
- key - klucz pozycji (bez parametrów)
- parameters - parametry klucza pozycji
- conditions - warunki oparte na grupie hostów i/lub tagach pozycji (w postaci wyrażenia)

Spacje są dozwolone wyłącznie wewnątrz wyrażenia warunków.

Użycie symbolu wieloznacznego

- Symbol wieloznaczny może być użyty do zastąpienia nazwy hosta, klucza pozycji lub pojedynczego parametru klucza pozycji.
- Albo host, albo klucz pozycji musi być określony bez symbolu wieloznacznego. Dlatego `/host/*` i `/*/key` są poprawnymi filtrami, ale `/*/*` jest nieprawidłowy.
- Symbol wieloznaczny nie może być użyty dla *części* nazwy hosta, klucza pozycji ani parametru klucza pozycji.
- Symbol wieloznaczny nie dopasowuje więcej niż jednego parametru klucza pozycji. Dlatego symbol wieloznaczny musi być określony osobno dla każdego parametru (tj. `key[abc,*,*]`).

Wyrażenie warunków

Wyrażenie warunków obsługuje:

- operandy:
  - group - grupa hostów
  - tag - tag pozycji
  - "<text>" - stała tekstowa, ze znakiem ucieczki \ do poprzedzania " i \
- operatory porównania ciągów znaków z rozróżnianiem wielkości liter: =, <>
- operatory logiczne: `and`, `or`, `not`
- grupowanie za pomocą nawiasów: ( )

Ujęcie stałych tekstowych w cudzysłów jest obowiązkowe. Obsługiwane jest tylko pełne porównanie ciągów znaków z rozróżnianiem wielkości liter.

Warning:
Podczas określania tagów w filtrze (tj. `tag="tagname:value"`), dwukropek ":" jest używany jako separator. Wszystko po nim jest traktowane jako wartość tagu. Dlatego obecnie nie jest obsługiwane określanie nazwy tagu zawierającej ":".

Przykłady

Można użyć złożonego filtra odwołującego się do klucza pozycji, grupy hostów i tagów, co ilustrują poniższe przykłady:

Przykład składni	Opis
`/host/key[abc,*]`	Dopasowuje podobne pozycje na tym hoście.
`/*/key`	Dopasowuje tę samą pozycję dowolnego hosta.
`/*/key?[group="ABC" and tag="tagname:value"]`	Dopasowuje tę samą pozycję dowolnego hosta z grupy ABC mającego tagi 'tagname:value'.
`/*/key[a,*,c]?[(group="ABC" and tag="Tag1") or (group="DEF" and (tag="Tag2" or tag="Tag3:value"))]`	Dopasowuje podobne pozycje dowolnego hosta z grupy ABC lub DEF z odpowiednimi tagami.

Wszystkie pozycje, do których istnieją odwołania, muszą istnieć i zbierać dane. W obliczeniach uwzględniane są tylko włączone pozycje na włączonych hostach. Pozycje w stanie nieobsługiwanym nie są uwzględniane.

> **Attention:**
> Jeśli klucz pozycji, do której istnieje odwołanie, zostanie zmieniony, filtr musi zostać zaktualizowany ręcznie.

Określenie nadrzędnej grupy hostów obejmuje grupę nadrzędną oraz wszystkie zagnieżdżone grupy hostów wraz z ich pozycjami.

Okres

Drugi parametr umożliwia określenie okresu agregacji. Okres można wyrazić wyłącznie jako czas; liczba wartości (poprzedzona znakiem **#**) nie jest obsługiwana.

W tym parametrze można używać obsługiwanych symboli jednostek, na przykład `5m` (pięć minut) zamiast `300s` (300 sekund) lub `1d` (jeden dzień) zamiast 86400 (86400 sekund).

W przypadku funkcji `last_foreach` okres jest parametrem opcjonalnym (obsługiwanym od wersji Zabbix 7.0), który można pominąć:

```
last_foreach(/*/key?[group="host group"])
```

Okres nie jest obsługiwany przez funkcję `exists_foreach`.

Dodatkowe parametry

bucket_rate_foreach

Trzeci opcjonalny parametr jest obsługiwany przez funkcję `bucket_rate_foreach`:

```
bucket_rate_foreach(item filter,time period,<parameter number>)
```

gdzie `<parameter number>` to pozycja wartości "bucket" w kluczu pozycji. Na przykład, jeśli wartość "bucket" w `myItem[aaa,0.2]` wynosi `0.2`, to jej pozycja to 2.

Domyślna wartość `<parameter number>` to 1.

count_foreach

Trzeci i czwarty opcjonalny parametr są obsługiwane przez funkcję `count_foreach`:

```
count_foreach(item filter,time period,<operator>,<pattern>)
```

Gdzie:

- **operator** to operator warunkowy dla wartości pozycji (musi być ujęty w podwójne cudzysłowy). Obsługiwane `operators`:
eq - równe
ne - różne
gt - większe
ge - większe lub równe
lt - mniejsze
le - mniejsze lub równe
like - dopasowuje, jeśli zawiera wzorzec (uwzględnia wielkość liter)
bitand - bitowe AND
regexp - dopasowanie z uwzględnieniem wielkości liter do wyrażenia regularnego podanego w `pattern`
iregexp - dopasowanie bez uwzględniania wielkości liter do wyrażenia regularnego podanego w `pattern`.
- **pattern** to wymagany wzorzec (argumenty typu string muszą być ujęte w podwójne cudzysłowy); obsługiwany, jeśli w trzecim parametrze określono *operator*.

Uwagi:

- Opcjonalnych parametrów *operator* lub *pattern* nie można pozostawić pustych po przecinku — można je tylko całkowicie pominąć.
- Przy *bitand* jako trzecim parametrze czwarty parametr `pattern` może być określony jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. `count_foreach()` oblicza "bitwise AND" z wartości i *maski*, a następnie porównuje wynik z *number_to_compare_with*. Jeśli wynik "bitwise AND" jest równy *number_to_compare_with*, wartość jest zliczana.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').
- Przy *regexp* lub *iregexp* jako trzecim parametrze czwarty parametr `pattern` może być zwykłym wyrażeniem regularnym lub globalnym (zaczynającym się od '@'). W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego. Na potrzeby dopasowania regexp wartości zmiennoprzecinkowe będą zawsze reprezentowane z 4 cyframi po przecinku '.'. Należy też pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę po przecinku.

Więcej informacji i przykładów użycia funkcji foreach można znaleźć w sekcji obliczenia agregujące.

Zachowanie zależne od dostępności

Poniższa tabela ilustruje, jak każda funkcja zachowuje się w przypadkach ograniczonej dostępności hosta/pozycji oraz danych historycznych.

Function	Disabled host	Unavailable host with data	Unavailable host without data	Disabled item	Unsupported item	Data retrieval error (SQL)
avg_foreach	ignoruj	zwróć średnią	ignoruj	ignoruj	ignoruj	ignoruj
bucket_rate_foreach	ignoruj	zwróć bucket rate	ignoruj	ignoruj	ignoruj	ignoruj
count_foreach	ignoruj	zwróć liczbę	0	ignoruj	ignoruj	ignoruj
exists_foreach	ignoruj	1	1	ignoruj	1	n/d
last_foreach	ignoruj	zwróć ostatnią wartość	ignoruj	ignoruj	ignoruj	ignoruj
max_foreach	ignoruj	zwróć maksimum	ignoruj	ignoruj	ignoruj	ignoruj
min_foreach	ignoruj	zwróć minimum	ignoruj	ignoruj	ignoruj	ignoruj
sum_foreach	ignoruj	zwróć sumę	ignoruj	ignoruj	ignoruj	ignoruj

Jeśli pozycja jest *ignorowana*, nic nie jest dodawane do agregacji.

Funkcje bitowe

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
bitand	Wartość ”bitwise AND” wartości pozycji i maski.
bitlshift	Bitowe przesunięcie w lewo wartości pozycji.
bitnot	Wartość ”bitwise NOT” wartości pozycji.
bitor	Wartość ”bitwise OR” wartości pozycji i maski.
bitrshift	Bitowe przesunięcie w prawo wartości pozycji.
bitxor	Wartość ”bitwise exclusive OR” wartości pozycji i maski.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Wyrażenia są akceptowane jako parametry
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >

bitand(value,mask)

Wartość operacji „bitwise AND” wartości pozycji i maski.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** — wartość do sprawdzenia;
- **mask** (wymagane) — 64-bitowa liczba całkowita bez znaku (0 - 18446744073709551615).

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu wykonuje się przez porównanie do 4, a nie 100.

Przykłady:

```
bitand(last(/host/key),12)=8 or bitand(last(/host/key),12)=4 #ustawiony 3. lub 4. bit, ale nie oba jednocz
bitand(last(/host/key),20)=16 #3. bit nie jest ustawiony, a 5. bit jest ustawiony
```

bitlshift(value,bits to shift)

Przesunięcie bitowe w lewo wartości pozycji.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **bits to shift** (wymagane) - liczba bitów do przesunięcia.

Chociaż porównanie jest wykonywane w sposób bitowy, wszystkie wartości muszą być podawane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie do 4, a nie 100.

bitnot(value)

Wartość operacji „bitwise NOT” dla wartości pozycji.
 Obsługiwane typy wartości: *Integer*.

Parametr:

- **value** — wartość do sprawdzenia.

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie z 4, a nie 100.

bitor(value,mask)

Wartość operacji „bitwise OR” dla wartości pozycji i maski.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **mask** (wymagane) - 64-bitowa liczba całkowita bez znaku (0 - 18446744073709551615).

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu wykonuje się przez porównanie z 4, a nie 100.

bitrshift(value,bits to shift)

Przesunięcie bitowe w prawo wartości pozycji.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **bits to shift** (wymagane) - liczba bitów do przesunięcia.

Chociaż porównanie jest wykonywane w sposób bitowy, wszystkie wartości muszą być podane i są zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie z 4, a nie 100.

bitxor(value,mask)

Wartość operacji „bitowe wykluczające LUB” dla wartości pozycji i maski.
 Obsługiwane typy wartości: *Integer*.

Parametry:

- **value** - wartość do sprawdzenia;
- **mask** (wymagany) - 64-bitowa liczba całkowita bez znaku (0 - 18446744073709551615).

Chociaż porównanie jest wykonywane bitowo, wszystkie wartości muszą być podawane i zwracane w postaci dziesiętnej. Na przykład sprawdzenie 3. bitu odbywa się przez porównanie z wartością 4, a nie 100.

Zobacz wszystkie obsługiwane funkcje.

Funkcje daty i czasu

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Attention:
Funkcji daty i czasu nie można używać samodzielnie w wyrażeniu; należy uwzględnić co najmniej jedną funkcję z innej grupy, odwołującą się do pozycji hosta (z wyjątkiem funkcji nodata()). Szczegółowe informacje o tym, jak funkcje daty i czasu działają w wyrażeniach, można znaleźć w sekcji Czas obliczeń.

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
date	Bieżąca data w formacie YYYYMMDD.
dayofmonth	Dzień miesiąca w zakresie od 1 do 31.

Function	Description
dayofweek	Dzień tygodnia w zakresie od 1 do 7.
now	Liczba sekund od Epoki (00:00:00 UTC, 1 stycznia 1970).
time	Bieżący czas w formacie HHMMSS.

Szczegóły funkcji

date

Bieżąca data w formacie YYYYMMDD.

Przykład:

`date()<20220101`

dayofmonth

Dzień miesiąca w zakresie od 1 do 31.

Przykład:

`dayofmonth()=1`

dayofweek

Dzień tygodnia w zakresie od 1 do 7 (pon. - 1, niedz. - 7).

Przykład (tylko dni robocze):

`dayofweek()<6`

Przykład (tylko weekend):

`dayofweek()>5`

now

Liczba sekund od Epoki (00:00:00 UTC, 1 stycznia 1970).

Przykład:

`now()<1640998800`

time

Bieżący czas w formacie HHMMSS.

Przykład (tylko pora nocna, 00:00-06:00):

`time()<060000`

Zobacz wszystkie obsługiwane funkcje.

Funkcje historii

Wszystkie wymienione tutaj funkcje są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby wyświetlić pełne informacje.

Funkcja	Opis
change	Wielkość różnicy między poprzednią a najnowszą wartością.
changecount	Liczba zmian między sąsiednimi wartościami w zdefiniowanym okresie oceny.
count	Liczba wartości w zdefiniowanym okresie oceny.
countunique	Liczba unikatowych wartości w zdefiniowanym okresie oceny.
find	Wyszukanie zgodnej wartości w zdefiniowanym okresie oceny.
first	Pierwsza (najstarsza) wartość w zdefiniowanym okresie oceny.
firstclock	Znacznik czasu pierwszej (najstarszej) wartości w zdefiniowanym okresie oceny.
fuzzytime	Sprawdzenie, jak bardzo czas pasywnego agent różni się od czasu serwera/proxy Zabbix.
jsonpath	Zwrócenie tablicy wyników zapytania JSONPath.

Funkcja	Opis
last	Najnowsza wartość.
lastclock	Znacznik czasu N-tej najnowszej wartości w zdefiniowanym okresie oceny.
logeventid	Sprawdzenie, czy identyfikator zdarzenia ostatniego wpisu dziennika pasuje do wyrażenia regularnego.
logseverity	Poziom ważności ostatniego wpisu dziennika.
logsource	Sprawdzenie, czy źródło ostatniego wpisu dziennika pasuje do wyrażenia regularnego.
logtimestamp	Znacznik czasu komunikatu dziennika dla N-tej najnowszej wartości pozycji dziennika.
monodec	Sprawdzenie, czy nastąpił monotoniczny spadek wartości.
monoinc	Sprawdzenie, czy nastąpił monotoniczny wzrost wartości.
nodata	Sprawdzenie, czy nie odebrano danych.
percentile	P-ty percentyl okresu, gdzie P (procent) jest określany przez trzeci parametr.
rate	Średnie tempo wzrostu na sekundę dla monotonicznie rosnącego licznika w zdefiniowanym okresie.
xmlxpath	Zwrócenie tablicy wyników zapytania XML XPath.

Wspólne parametry

- `/host/key` jest wspólnym wymaganym pierwszym parametrem funkcji odwołujących się do historii pozycji hosta.
- `(sec|#num)<:time shift>` jest wspólnym drugim parametrem funkcji odwołujących się do historii pozycji hosta, gdzie:
  - **sec** - maksymalny okres oceny w sekundach (można używać przyrostków czasu), lub
  - **#num** - maksymalny zakres oceny w najnowszych zebranych wartościach (jeśli poprzedzony znakiem hash).
  - **time shift** (opcjonalny) umożliwia przesunięcie punktu oceny wstecz w czasie. Zobacz więcej szczegółów na temat określania przesunięcia w czasie.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone za pomocą < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `(sec|#num)<:time shift>` nigdy nie mogą być ujmowane w cudzysłów

change(/host/key)

Wielkość różnicy między poprzednią a najnowszą wartością.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.
 Dla ciągów znaków zwraca: 0 - wartości są równe; 1 - wartości różnią się.

Parametry: zobacz wspólne parametry.

Uwagi:

- Zostanie obliczona różnica liczbowa, jak pokazano na poniższych przykładowych wartościach wejściowych („poprzednia” i „najnowsza” wartość = różnica):
’1’ i ’5’ = +4
’3’ i ’1’ = −2
’0’ i ’-2.5’ = −2.5.
- W wyzwalaczach funkcja jest przydatna do wykrywania nagłych skoków (lub spadków), resetów liczników oraz przejść stanów liczbowych.
- Zobacz także: abs do porównania.

Przykłady:

```
change(/host/system.uptime)<0 #zmiana czasu działania systemu była ujemna od ostatniej wartości (wskazuje
change(/host/system.cpu.load[all,avg1])>2 #obciążenie CPU (w ciągu jednej minuty) wzrosło o więcej niż 2 c
change(/host/vfs.fs.size[/,free])<-1G #ilość wolnego miejsca na dysku spadła o więcej niż 1 GB między kole
```

changecount(/host/key,(sec|#num)<:time shift>,<mode>)

Liczba zmian między sąsiednimi wartościami w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry
- **mode** (musi być ujęty w podwójny cudzysłów) — możliwe wartości: *all* — zliczaj wszystkie zmiany (domyślnie); *dec* — zliczaj spadki; *inc* — zliczaj wzrosty

Uwagi:

- Dla nienumerycznych typów wartości parametr *mode* jest ignorowany.
- W wyzwalaczach przydatne do wykrywania częstych zmian stanu (wskazujących na niestabilność).

Przykłady:

```
changecount(/host/icmpping,10m)>5 #status ping zmienił się więcej niż 5 razy w ciągu 10 minut
changecount(/host/vfs.file.contents["/sys/class/net/eth0/operstate"],1h)>5 #stan operacyjny eth0 zmienił s
changecount(/host/proc.num[httpd],15m)>10 #liczba procesów httpd zmieniła się więcej niż 10 razy w ciągu 1
changecount(/host/key,#10,"inc") #liczba wzrostów wartości (względem wartości sąsiedniej) wśród ostatnich
changecount(/host/key,24h,"dec") #liczba spadków wartości (względem wartości sąsiedniej) w ciągu ostatnich
```

count(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Liczba wartości w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz common parameters
- **operator** (musi być ujęty w podwójne cudzysłowy) Obsługiwane operators:
eq - równe (domyślnie dla integer, float)
ne - nierówne
gt - większe niż
ge - większe lub równe
lt - mniejsze niż
le - mniejsze lub równe
like (domyślnie dla string, text, log) - dopasowuje, jeśli zawiera wzorzec (rozróżnia wielkość liter)
bitand - bitowe AND
regexp - dopasowanie wyrażenia regularnego podanego w pattern z uwzględnieniem wielkości liter
iregexp - dopasowanie wyrażenia regularnego podanego w pattern bez uwzględniania wielkości liter
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójne cudzysłowy)

Uwagi:

- Pozycje typu Float są dopasowywane z dokładnością 2.22e-16.
- *like* nie jest obsługiwany jako operator dla wartości typu integer.
- *like* i *bitand* nie są obsługiwane jako operatory dla wartości typu float.
- Dla wartości typu string, text i log obsługiwane są tylko operatory *eq*, *ne*, *like*, *regexp* i *iregexp*.
- Przy operatorze *bitand* czwarty parametr pattern może być określony jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. count() oblicza "bitowe AND" z wartości i *maski*, a następnie porównuje wynik z *number_to_compare_with*. Jeśli wynik "bitowego AND" jest równy *number_to_compare_with*, wartość jest liczona.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').
- Przy operatorze *regexp* lub *iregexp* czwarty parametr pattern może być zwykłym wyrażeniem regularnym lub globalnym (zaczynającym się od '@') wyrażeniem regularnym. W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego. Na potrzeby dopasowania regexp wartości typu float będą zawsze reprezentowane z 4 cyframi po przecinku. Należy też pamiętać, że dla dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę po przecinku.

Przykłady:

```
count(/host/icmpping,30m,,"0")>5 #ping nie powiódł się więcej niż 5 razy w ciągu 30 minut
count(/host/key,10m,"like","error") #liczba wartości z ostatnich 10 minut do teraz, które zawierają 'error
count(/host/key,10m,,12) #liczba wartości z ostatnich 10 minut do teraz, które są równe '12'
count(/host/key,10m,"gt",12) #liczba wartości z ostatnich 10 minut do teraz, które są większe niż '12'
count(/host/key,#10,"gt",12) #liczba wartości w ostatnich 10 wartościach do teraz, które są większe niż '1
count(/host/key,10m:now-1d,"gt",12) #liczba wartości między 24 godzinami i 10 minutami a 24 godzinami temu
count(/host/key,10m,"bitand","6/7") #liczba wartości z ostatnich 10 minut do teraz, które mają '110' (w sy
count(/host/key,10m:now-1d) #liczba wartości między 24 godzinami i 10 minutami a 24 godzinami temu od tera
```

countunique(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Liczba unikalnych wartości w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz common parameters
- **operator** (musi być ujęty w podwójne cudzysłowy). Obsługiwane operators:
eq - równe (domyślnie dla integer, float)
ne - różne
gt - większe
ge - większe lub równe
lt - mniejsze
le - mniejsze lub równe
like (domyślnie dla string, text, log) - dopasowuje, jeśli zawiera wzorzec (uwzględnia wielkość liter)
bitand - bitowe AND
regexp - dopasowanie wyrażenia regularnego podanego w pattern z uwzględnieniem wielkości liter
iregexp - dopasowanie wyrażenia regularnego podanego w pattern bez uwzględniania wielkości liter
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójne cudzysłowy)

Uwagi:

- Pozycje typu Float są dopasowywane z dokładnością 2.22e-16.
- *like* nie jest obsługiwany jako operator dla wartości typu integer.
- *like* i *bitand* nie są obsługiwane jako operatory dla wartości typu float.
- Dla wartości typu string, text i log obsługiwane są tylko operatory *eq*, *ne*, *like*, *regexp* i *iregexp*.

- Przy operatorze *bitand* czwarty parametr `pattern` może być określony jako dwie liczby rozdzielone znakiem '/': **number_to_compare_with/mask**. countunique() oblicza "bitwise AND" z wartości i *maski* oraz porównuje wynik z *number_to_compare_with*. Jeśli wynik "bitwise AND" jest równy *number_to_compare_with*, wartość jest liczona.
Jeśli *number_to_compare_with* i *mask* są równe, wystarczy podać tylko *mask* (bez '/').
- Przy operatorze *regexp* lub *iregexp* czwarty parametr `pattern` może być zwykłym wyrażeniem regularnym lub global wyrażeniem regularnym (zaczynającym się od '@'). W przypadku globalnych wyrażeń regularnych uwzględnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego. Na potrzeby dopasowania regexp wartości typu float będą zawsze reprezentowane z 4 cyframi po przecinku '.'. Należy też pamiętać, że w przypadku dużych liczb różnica między reprezentacją dziesiętną (przechowywaną w bazie danych) a binarną (używaną przez serwer Zabbix) może wpływać na 4. cyfrę po przecinku.

Przykłady:

```
countunique(/host/key,10m) #liczba unikalnych wartości z ostatnich 10 minut do teraz
countunique(/host/key,10m,"like","error") #liczba unikalnych wartości z ostatnich 10 minut do teraz, które
countunique(/host/key,10m,,12) #liczba unikalnych wartości z ostatnich 10 minut do teraz, które są równe '
countunique(/host/key,10m,"gt",12) #liczba unikalnych wartości z ostatnich 10 minut do teraz, które są wię
countunique(/host/key,#10,"gt",12) #liczba unikalnych wartości z ostatnich 10 wartości do teraz, które są
countunique(/host/key,10m:now-1d,"gt",12) #liczba unikalnych wartości między 24 godzinami a 10 minutami or
countunique(/host/key,10m,"bitand","6/7") #liczba unikalnych wartości z ostatnich 10 minut do teraz, mając
countunique(/host/key,10m:now-1d) #liczba unikalnych wartości między 24 godzinami a 10 minutami oraz 24 go
```

find(/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Znajduje dopasowanie wartości w zdefiniowanym okresie oceny.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.
 Zwraca: 1 - znaleziono; 0 - w przeciwnym razie.

Parametry:

- Zobacz common parameters
- **sec** lub **#num** (opcjonalnie) - domyślnie używana jest najnowsza wartość, jeśli nie zostanie określona
- **operator** (musi być ujęty w podwójne cudzysłowy) Obsługiwane `operators`:
eq - równe (domyślnie dla integer, float)
ne - nierówne
gt - większe
ge - większe lub równe
lt - mniejsze
le - mniejsze lub równe
like (domyślnie dla string, text, log) - dopasowuje, jeśli zawiera ciąg podany w `pattern` (uwzględnia wielkość liter)
bitand - bitowe AND
regexp - dopasowanie wyrażenia regularnego podanego w `pattern` z uwzględnieniem wielkości liter
iregexp - dopasowanie wyrażenia regularnego podanego w `pattern` bez uwzględniania wielkości liter
- **pattern** - wymagany wzorzec (argumenty tekstowe muszą być ujęte w podwójne cudzysłowy); wyrażenie regularne Perl Compatible Regular Expression (PCRE), jeśli `operator` to *regexp*, *iregexp*

Uwagi:

- Jeśli przetwarzanych jest więcej niż jedna wartość, zwracane jest '1', jeśli istnieje co najmniej jedna pasująca wartość.
- *like* nie jest obsługiwany jako operator dla wartości typu integer.
- *like* i *bitand* nie są obsługiwane jako operatory dla wartości typu float.
- Dla wartości typu string, text i log obsługiwane są tylko operatory *eq*, *ne*, *like*, *regexp* i *iregexp*.
- W przypadku użycia *regexp* lub *iregexp* jako operatora, czwarty parametr `pattern` może być zwykłym wyrażeniem regularnym lub globalnym (zaczynającym się od '@').
  W przypadku globalnych wyrażeń regularnych rozróżnianie wielkości liter jest dziedziczone z ustawień globalnego wyrażenia regularnego.

Przykłady:

```
find(/host/key,10m,"like","error") #znajduje wartość zawierającą 'error' w ciągu ostatnich 10 minut do ter
find(/host/agent.version,,"like","beta")=1 #agent Zabbix ma wersję beta, musi zostać zaktualizowany
find(/host/log[/var/log/nginx/access.log],,"regexp"," 500 ")=1 #został znaleziony wewnętrzny błąd serwera
```

first(/host/key,sec<:time shift>)

Pierwsza (najstarsza) wartość w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry

Zobacz także last().

Przykład:

```
first(/host/key,1h) #retrieve the oldest value within the last hour until now
```

firstclock(/host/key,sec<:time shift>)

Znacznik czasu najstarszej wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry.

Funkcja kończy się błędem, jeśli w podanym okresie nie zebrano żadnych danych.

Zobacz także lastclock().

Przykłady:

```
firstclock(/host/key,1h) #pobierz znacznik czasu najstarszej wartości z ostatniej godziny
firstclock(/host/key,1h:now-24h) #pobierz znacznik czasu najstarszej wartości z ostatniej godziny sprzed d
```

fuzzytime(/host/key,sec)

Sprawdza, o ile czas pasywnego agenta różni się od czasu serwera/proxy Zabbixa.
 Obsługiwane typy wartości: *Float*, *Integer*.
 Zwraca: 1 - różnica między wartością pozycji pasywnej (jako znacznik czasu) a znacznikiem czasu serwera/proxy Zabbixa (czasem zbierania wartości) jest mniejsza lub równa *sec* sekund; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry.

Uwagi:

- Zwykle używane z pozycją 'system.localtime' do sprawdzenia, czy czas lokalny jest zsynchronizowany z czasem lokalnym serwera Zabbixa. *Uwaga*, że 'system.localtime' musi być skonfigurowane jako sprawdzenie pasywne dla agenta Zabbixa; w przypadku agenta Zabbix 2 może być skonfigurowane jako sprawdzenie aktywne.
- Może być również używane z kluczem `vfs.file.time[/path/file,modify]` do sprawdzenia, czy plik nie był aktualizowany przez długi czas.
- Ta funkcja nie jest zalecana do użycia w złożonych wyrażeniach wyzwalaczy (z udziałem wielu pozycji), ponieważ może powodować nieoczekiwane wyniki (różnica czasu będzie mierzona na podstawie najnowszej metryki), np. w `fuzzytime(/Host/system.localtime,60s)=0 or last(/Host/trap)<>0`.

Przykłady:

```
fuzzytime(/host/system.localtime,5m)=0 #czas lokalny klienta różni się od czasu serwera/proxy Zabbixa o wi
fuzzytime(/host/system.localtime,5m)=0 and nodata(/host/system.localtime,10m)=0 #czas lokalny klienta różn
```

jsonpath(/host/key,(sec|#num)<:time shift>,pattern)

Zwraca tablicę wyników zapytań JSONPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry
- **pattern** - wymagany wzorzec (argumenty typu string muszą być ujęte w podwójny cudzysłów)

Uwagi:

- Ta funkcja jest używana z funkcjami agregującymi.
- Zapytanie JSONPath jest wykonywane dla każdej wybranej wartości historycznej; każdy niepusty wynik pomyślnego zapytania jest dodawany do tablicy. Jeśli wynik zapytania dla pojedynczej wartości historycznej zawiera wiele dopasowań, każde dopasowanie jest dodawane do tablicy osobno.
- Funkcja kończy działanie błędem, jeśli:
  - Format `pattern` jest nieprawidłowy.
  - Dowolna z wartości historycznych zawiera nieprawidłowy JSON lub zapytanie JSONPath kończy się niepowodzeniem dla tej wartości.
- Funkcja zwraca pustą tablicę, jeśli:
  - Nie wybrano żadnych wartości historycznych.
  - Wszystkie wybrane wartości historyczne zawierają nieprawidłowy JSON.

Przykład:

```
max(jsonpath(/host/http.response,5m,"$.response_time")) > 500
```

last(/host/key,<#num<:time shift»)

Najnowsza wartość.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry
- **#num** (opcjonalnie) — N-ta najnowsza wartość

Uwagi:

- Należy zwrócić uwagę, że okres czasu oznaczony hashem (#N) działa tutaj inaczej niż w przypadku wielu innych funkcji. Na przykład: `last(/host/key)` jest zawsze równe `last(/host/key,#1)`; `last(/host/key,#3)` — trzecia najnowsza wartość (*nie* trzy najnowsze wartości).
- Zabbix nie gwarantuje dokładnej kolejności wartości, jeśli w historii istnieje więcej niż dwie wartości w ciągu jednej sekundy.
- Zobacz także first().

Przykłady:

```
last(/host/key) #retrieve the last value
last(/host/key,#2) #retrieve the previous value
last(/host/key,#1)<>last(/host/key,#2) #last two values differ
```

lastclock(/host/key,<#num<:time shift»)

Znacznik czasu N-tej najnowszej wartości w zdefiniowanym okresie obliczania.
 Obsługiwane typy wartości: *Float*, *Integer*, *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry;
- **#num** (opcjonalnie) — N-ta najnowsza wartość.

Funkcja kończy się błędem, jeśli w podanym okresie nie zebrano żadnych danych lub nie zebrano N-tej wartości.

Zobacz także firstclock().

Przykłady:

```
lastclock(/host/key) #pobierz znacznik czasu najnowszej wartości
lastclock(/host/key,#2) #pobierz znacznik czasu 2. najnowszej wartości
now()-lastclock(/host/heartbeat.item)>300 #minęło więcej niż 5 minut od aktualizacji pozycji heartbeat
lastclock(/host/system.cpu.load[all,avg1])-lastclock(/host/system.cpu.load[all,avg1],#2)>300 #interwał akt
((now()-lastclock(/host/system.cpu.load[all,avg1])<120 and last(/host/system.cpu.load[all,avg1])>5)) #wart
```

logeventid(/host/key,<#num<:time shift»,<pattern>)

Sprawdza, czy identyfikator zdarzenia ostatniego wpisu dziennika pasuje do wyrażenia regularnego.
 Obsługiwane typy wartości: *Log*.
 Zwraca: 0 - nie pasuje; 1 - pasuje.

Parametry:

- Zobacz common parameters
- **#num** (opcjonalnie) - N-ta najnowsza wartość
- **pattern** (opcjonalnie) - wyrażenie regularne opisujące wymagany wzorzec, w stylu Perl Compatible Regular Expression (PCRE) (argumenty tekstowe muszą być ujęte w podwójne cudzysłowy)

Przykłady:

```
logeventid(/host/eventlog[Security],,"^4625$")=1 #wpis dziennika z identyfikatorem pasującym do "4625" (ni
logeventid(/host/eventlog[System],,"^(6008|41)$")=1 #wpis dziennika z identyfikatorem pasującym do "6008"
```

logseverity(/host/key,<#num<:time shift»)

Poziom ważności ostatniego wpisu logu.
 Obsługiwane typy wartości: *Log*.
 Zwraca: 0 - domyślny poziom ważności; N - poziom ważności (liczba całkowita, przydatne dla dzienników zdarzeń Windows: 1 - Informacja, 2 - Ostrzeżenie, 4 - Błąd, 7 - Audyt niepowodzeń, 8 - Audyt powodzeń, 9 - Krytyczny, 10 - Szczegółowy).

Parametry:

- Zobacz wspólne parametry
- **#num** (opcjonalnie) - N-ta najnowsza wartość

Zabbix pobiera poziom ważności logu z pola **Information** dziennika zdarzeń Windows.

Przykłady:

```
logseverity(/host/log[/var/log/syslog],10m)>3 #a log entry with severity above "3" found
logseverity(/host/eventlog[System],10m)=4 #a log entry with severity equaling "Error" found
```

logsource(/host/key,<#num<:time shift»,<pattern>)

Sprawdza, czy źródło logu ostatniego wpisu logu pasuje do wyrażenia regularnego.
 Obsługiwane typy wartości: *Log*.
 Zwraca: 0 - nie pasuje; 1 - pasuje.

Parametry:

- Zobacz common parameters
- **#num** (opcjonalnie) - N-ta najnowsza wartość
- **pattern** (opcjonalnie) - wyrażenie regularne opisujące wymagany wzorzec, w stylu Perl Compatible Regular Expression (PCRE) (argumenty tekstowe muszą być ujęte w podwójne cudzysłowy)

Zwykle używane dla dzienników zdarzeń systemu Windows.

Przykłady:

```
logsource(/host/eventlog[Application],,"MSSQLSERVER")=1 #znaleziono wpis logu ze źródłem pasującym do "MSS
logsource(/host/eventlog[System],,"Service Control Manager")=1 #znaleziono wpis logu ze źródłem pasującym
logsource(/host/eventlog[System],,"Service Control Manager")=1 and logeventid(/host/eventlog[System],,"^70
```

logtimestamp(/host/key,<#num<:time shift»)

Znacznik czasu komunikatu dziennika dla N-tej najnowszej wartości pozycji dziennika.
 Obsługiwane typy wartości: *Log*.

Parametry:

- Zobacz wspólne parametry;
- **#num** (opcjonalnie) — N-ta najnowsza wartość.

Uwagi:

- Obliczanie przesunięcia czasu opiera się na zegarze wartości pozycji, a nie na znaczniku czasu komunikatu dziennika;
- Funkcja kończy się błędem, jeśli:
  - odebrano pozycję typu innego niż log;
  - w podanym okresie nie zebrano żadnych danych;
  - nie zebrano N-tej wartości;
  - komunikat dziennika nie zawiera wartości znacznika czasu.

Przykłady:

```
logtimestamp(/host/key) #retrieve the timestamp of the latest log message
logtimestamp(/host/key,#2) #retrieve the timestamp of the 2nd latest log message
logtimestamp(/host/key,#3:now-1d) #retrieve the timestamp of the 3rd latest log message a day ago
```

monodec(/host/key,(sec|#num)<:time shift>,<mode>)

Sprawdza, czy nastąpił monotoniczny spadek wartości.
 Obsługiwane typy wartości: *Integer*.
 Zwraca: 1 - jeśli wszystkie elementy w okresie czasu stale maleją; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry
- **mode** (musi być ujęty w podwójny cudzysłów) - *weak* (każda wartość jest mniejsza lub taka sama jak poprzednia; domyślnie) lub *strict* (każda wartość zmalała)

Przykłady:

```
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 #liczba procesów nginx monotonicznie malała przez 10 minut
```

monoinc(/host/key,(sec|#num)<:time shift>,<mode>)

Sprawdza, czy wystąpił monotoniczny wzrost wartości.
 Obsługiwane typy wartości: *Integer*.
 Zwraca: 1 - jeśli wszystkie elementy w okresie czasu stale rosną; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry
- **mode** (musi być ujęty w podwójny cudzysłów) - *weak* (każda wartość jest większa lub taka sama jak poprzednia; domyślnie) lub *strict* (każda wartość wzrosła)

Przykłady:

```
monoinc(/host/system.localtime,#3,"strict")=0 #lokalny czas systemowy nie wzrastał w sposób ciągły
monoinc(/host/vfs.dir.count[/mnt/data/logs],24h,"weak")=0 #wyzwalacz, jeśli liczba plików pozostawała bez
```

nodata(/host/key,sec,<mode>)

Sprawdza, czy nie odebrano żadnych danych.
 Obsługiwane typy wartości: *Integer*, *Float*, *Character*, *Text*, *Log*.
 Zwraca: 1 - jeśli w zdefiniowanym okresie nie odebrano żadnych danych; 0 - w przeciwnym razie.

Parametry:

- Zobacz wspólne parametry
- **sec** - okres nie powinien być krótszy niż 30 sekund, ponieważ proces synchronizatora historii oblicza tę funkcję tylko co 30 sekund; `nodata(/host/key,0)` jest niedozwolone
- **mode** - jeśli ustawiono wartość *strict* (w podwójnych cudzysłowach), funkcja będzie niewrażliwa na dostępność proxy (szczegółowe informacje znajdują się w komentarzach)

Komentarze:

- wyzwalacze „nodata" monitorowane przez proxy są domyślnie wrażliwe na dostępność proxy - jeśli proxy stanie się niedostępne, wyzwalacze „nodata" nie zostaną uruchomione natychmiast po przywróceniu połączenia, lecz pominą dane z opóźnionego okresu. Należy pamiętać, że w przypadku pasywnych proxy tłumienie jest aktywowane, jeśli połączenie zostanie przywrócone po upływie więcej niż 15 sekund i nie później niż 2 sekundy. W przypadku aktywnych proxy tłumienie jest aktywowane, jeśli połączenie zostanie przywrócone po upływie więcej niż 15 sekund. Aby wyłączyć wrażliwość na dostępność proxy, użyj trzeciego parametru, np.: `nodata(/host/key,5m,"strict")`; w takim przypadku funkcja zostanie uruchomiona, gdy tylko upłynie okres oceny (pięć minut) bez danych.
- Ta funkcja wyświetli błąd, jeśli w okresie określonym przez 1. parametr:
- nie ma danych i serwer Zabbix został uruchomiony ponownie
- nie ma danych i zakończył się okres konserwacji
- nie ma danych, a pozycja została dodana lub ponownie włączona.
- Błędy są wyświetlane w kolumnie *Info* na liście wyzwalaczy.
- Ta funkcja może nie działać prawidłowo, jeśli między serwerem Zabbix, proxy i agentem występują różnice czasu. Zobacz także: Wymagania dotyczące synchronizacji czasu.

Przykład:

```
nodata(/host/agent.ping,5m)=1 #wyzwalacz, jeśli przez 5 minut nie odebrano danych od agenta Zabbix
```

percentile(/host/key,(sec|#num)<:time shift>,percentage)

P-ty percentyl okresu, gdzie P (percentage) jest określony przez trzeci parametr.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry
- **percentage** - liczba zmiennoprzecinkowa z zakresu od 0 do 100 (włącznie), z maksymalnie 4 cyframi po przecinku

Przykłady:

```
percentile(/host/net.if.in[eth0,bytes],1h,95)>1000000 #95. percentyl wejściowego ruchu sieciowego (bajty/s
percentile(/host/system.cpu.util,5m,95)>80 #95. percentyl procentowego wykorzystania CPU w czasie użytkown
percentile(/host/icmppingsec[192.168.0.2],15m,95)>0.15 #większość pomiarów opóźnienia jest poniżej 150 ms,
percentile(/host/net.if.in[eth0,bytes],1h,50) #oblicz 50. percentyl (wartość mediany) przychodzącego ruchu
(percentile(/host/net.if.in[eth0,bytes],1h,50)+percentile(/host/net.if.in[eth0,bytes],1h,51))/2 #oblicz do
```

rate(/host/key,sec<:time shift>)

Średni przyrost na sekundę w monotonicznie rosnącym liczniku w zdefiniowanym okresie czasu.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry

Funkcjonalnie odpowiada funkcji 'rate' w PromQL.

Przykłady:

```
rate(/host/key,30s) #jeśli monotoniczny przyrost w ciągu 30 sekund wynosi 20, ta funkcja zwraca 0.67.
rate(/host/net.if.in[eth0,bytes],5m)>500000 #szybkość ruchu przychodzącego interfejsu na eth0 przekroczyła
rate(/host/app.requests.count,1m)>100 #licznik żądań wzrósł do >100 żądań na sekundę w ostatniej minucie
```

xmlxpath(/host/key,(sec|#num)<:time shift>,pattern)

Zwraca tablicę wyników zapytań XML XPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- Zobacz wspólne parametry

- **pattern** - wymagany wzorzec (argumenty tekstowe muszą być ujęte w podwójny cudzysłów)

Uwagi:

- Tej funkcji używa się z funkcjami agregującymi.
- Zapytanie XML XPath jest wykonywane dla każdej wybranej wartości historycznej; każdy niepusty, pomyślnie uzyskany wynik zapytania jest dodawany do tablicy. Jeśli wynik zapytania dla pojedynczej wartości historycznej zawiera wiele dopasowań, każde dopasowanie jest dodawane do tablicy osobno.
- Funkcja kończy działanie błędem, jeśli:
  - Format `pattern` jest nieprawidłowy.
  - Zabbix został skompilowany bez obsługi XML.
  - Dowolna z wartości historycznych zawiera nieprawidłowy kod XML lub zapytanie XPath kończy się niepowodzeniem dla tej wartości.
- Funkcja zwraca pustą tablicę, jeśli:
  - Nie wybrano żadnych wartości historycznych.
  - Wszystkie wybrane wartości historyczne zawierają nieprawidłowy kod XML.

Przykład:

```
max(xmlxpath(/host/http.response,5m,"/response/response_time")) > 500
```

Zobacz wszystkie obsługiwane funkcje.

Funkcje trendów

Funkcje trendów, w przeciwieństwie do funkcji historii, wykorzystują dane trend do obliczeń.

Trendy przechowują godzinowe wartości agregowane. Funkcje trendów używają tych godzinowych średnich, dzięki czemu są przydatne do długoterminowej analizy.

Wyniki funkcji trendów są buforowane, więc wielokrotne wywołania tej samej funkcji z tymi samymi parametrami pobierają informacje z bazy danych tylko raz. Bufor funkcji trendów jest kontrolowany przez parametr serwera TrendFunctionCacheSize.

Użyj `zabbix[tcache,cache,paccessed]`, aby oszacować, jaka część bufora jest faktycznie używana. Niskie `paccessed`, wraz z niskimi `pitems` i `pmisses`, zwykle oznacza, że rozmiar bufora można zmniejszyć.

Wyzwalacze, które odwołują się **wyłącznie** do funkcji trendów, są oceniane raz na najmniejszy okres czasu w wyrażeniu. Na przykład wyzwalacz taki jak

```
trendavg(/host/key,1d:now/d) > 1 or trendavg(/host/key2,1w:now/w) > 2
```

będzie oceniany raz dziennie. Jeśli wyzwalacz zawiera zarówno funkcje trendów, jak i historii (lub daty i czasu i/lub nodata()), jest obliczany zgodnie ze zwykłymi zasadami.

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Pozycjach obliczanych
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
baselinedev	Zwraca liczbę odchyleń (według algorytmu stddevpop) między ostatnim okresem danych a tymi samymi okresami danych w poprzednich sezonach.
baselinewma	Oblicza wartość bazową przez uśrednianie danych z tego samego przedziału czasu w wielu równych okresach czasu („sezonach”) przy użyciu algorytmu ważonej średniej ruchomej.
trendavg	Średnia wartości trendów w zdefiniowanym okresie czasu.
trendcount	Liczba pomyślnie pobranych wartości historii użytych do obliczenia wartości trendu w zdefiniowanym okresie czasu.
trendmax	Maksimum wartości trendów w zdefiniowanym okresie czasu.
trendmin	Minimum wartości trendów w zdefiniowanym okresie czasu.
trendstl	Zwraca współczynnik anomalii w okresie wykrywania - wartość dziesiętną między 0 a 1, równą `((liczba wartości anomalii)/(łączna liczba wartości))`.
trendsum	Suma wartości trendów w zdefiniowanym okresie czasu.

Wspólne parametry

- `/host/key` to wspólny obowiązkowy pierwszy parametr
- `time period:time shift` to wspólny drugi parametr, gdzie:
  - **time period** - okres czasu (minimum „1h”), zdefiniowany jako <N><time unit>, gdzie `N` - liczba jednostek czasu, `time unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok).
  - **time shift** - przesunięcie okresu czasu (zobacz przykłady funkcji)

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczane za pomocą < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `time period:time shift` nigdy nie mogą być ujmowane w cudzysłów

baselinedev(/host/key,data period:time shift,season unit,num seasons)

Zwraca liczbę odchyleń (według algorytmu stddevpop) między ostatnim okresem danych a tymi samymi okresami danych w poprzednich sezonach.

Parametry:

- Zobacz wspólne parametry
- **data period** - okres zbierania danych w ramach sezonu, zdefiniowany jako <N><time unit>, gdzie:
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok), musi być równa sezonowi lub od niego mniejsza
- **season unit** - jednostka kalendarzowa definiująca jeden sezon (h, d, w, M, y); nie może być mniejsza niż data period
- **num seasons** - liczba sezonów do oceny

Przykłady:

```
baselinedev(/host/key,1d:now/d,"M",6) #obliczanie liczby odchyleń standardowych (populacyjnych) między pop
baselinedev(/host/key,1h:now/h,"d",10)  #obliczanie liczby populacyjnych odchyleń standardowych między pop
```

baselinewma(/host/key,data period:time shift,season unit,num seasons)

Oblicza linię bazową przez uśrednianie danych z tego samego przedziału czasowego w wielu równych okresach czasu („sezonach”) przy użyciu algorytmu ważonej średniej ruchomej.

Parametry:

- Zobacz wspólne parametry
- **data period** - okres zbierania danych w ramach sezonu, zdefiniowany jako <N><jednostka czasu>, gdzie:
`N` - liczba jednostek czasu
`jednostka czasu` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok), musi być równa sezonowi lub od niego mniejsza
Time shift - przesunięcie czasowe, określa przesunięcie okresu czasu, definiuje koniec przedziału zbierania danych w sezonach (zobacz przykłady)
- **season unit** - jednostka kalendarzowa definiująca jeden sezon (h, d, w, M, y); nie może być mniejsza niż data period
- **num seasons** - liczba sezonów do oceny

Przykłady:

```
baselinewma(/host/key,1h:now/h,"d",3) #obliczanie linii bazowej dla tej samej godziny dnia w ostatnich 3 p
baselinewma(/host/key,2h:now/h,"d",3) #obliczanie linii bazowej dla tego samego dwugodzinnego przedziału c
baselinewma(/host/key,1d:now/d,"M",4) #obliczanie linii bazowej dla tego samego dnia miesiąca co „wczoraj"
```

trendavg(/host/key,time period:time shift)

Średnia wartości trendów w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendavg(/host/key,1h:now/h) #średnia dla poprzedniej godziny (np. 12:00-13:00)
trendavg(/host/key,1h:now/h-1h) #średnia sprzed dwóch godzin (11:00-12:00)
trendavg(/host/key,1h:now/h-2h) #średnia sprzed trzech godzin (10:00-11:00)
trendavg(/host/key,1M:now/M-1y) #średnia dla poprzedniego miesiąca rok temu
```

trendcount(/host/key,time period:time shift)

Liczba pomyślnie pobranych wartości historii użytych do obliczenia wartości trendu w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendcount(/host/key,1h:now/h) #liczba wartości dla poprzedniej godziny (np. 12:00-13:00)
trendcount(/host/key,1h:now/h-1h) #liczba wartości sprzed dwóch godzin (11:00-12:00)
trendcount(/host/key,1h:now/h-2h) #liczba wartości sprzed trzech godzin (10:00-11:00)
trendcount(/host/key,1M:now/M-1y) #liczba wartości dla poprzedniego miesiąca rok temu
```

trendmax(/host/key,time period:time shift)

Maksimum w wartościach trendu w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendmax(/host/key,1h:now/h) #maksimum dla poprzedniej godziny (np. 12:00-13:00)
trendmax(/host/key,1h:now/h) - trendmin(/host/key,1h:now/h) → oblicza różnicę między wartościami maksymaln
trendmax(/host/key,1h:now/h-1h) #maksimum sprzed dwóch godzin (11:00-12:00)
trendmax(/host/key,1h:now/h-2h) #maksimum sprzed trzech godzin (10:00-11:00)
trendmax(/host/key,1M:now/M-1y) #maksimum dla poprzedniego miesiąca rok temu
```

trendmin(/host/key,time period:time shift)

Minimum w wartościach trendu w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendmin(/host/key,1h:now/h) #minimum dla poprzedniej godziny (np. 12:00-13:00)
trendmax(/host/key,1h:now/h) - trendmin(/host/key,1h:now/h) → oblicza różnicę między wartościami maksymaln
trendmin(/host/key,1h:now/h-1h) #minimum sprzed dwóch godzin (11:00-12:00)
trendmin(/host/key,1h:now/h-2h) #minimum sprzed trzech godzin (10:00-11:00)
trendmin(/host/key,1M:now/M-1y) #minimum dla poprzedniego miesiąca rok temu
```

trendstl(/host/key,eval period:time shift,detection period,season,<deviations>,<devalg>,<s window>)

Zwraca współczynnik anomalii w okresie wykrywania — wartość dziesiętną z zakresu od 0 do 1, obliczaną jako `((liczba wartości anomalii)/(całkowita liczba wartości))`.

Parametry:

- Zobacz wspólne parametry
- **eval period** - okres czasu, który ma zostać poddany dekompozycji (minimum „1h”), zdefiniowany jako <N><time unit>, gdzie
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok)
- **detection period** - okres czasu przed końcem eval period, dla którego obliczane są anomalie (minimum „1h”, nie może być dłuższy niż eval period), zdefiniowany jako <N><time unit>, gdzie
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień)
- **season** - najkrótszy okres czasu, w którym oczekiwany jest powtarzający się wzorzec („sezon”) (minimum „2h”, nie może być dłuższy niż eval period, liczba wpisów w eval period musi być większa niż dwukrotność wynikowej częstotliwości (season/h)), zdefiniowany jako <N><time unit>, gdzie
`N` - liczba jednostek czasu
`time unit` - h (godzina), d (dzień), w (tydzień)
- **deviations** - liczba odchyleń (obliczana przez devalg), przy której wartość jest uznawana za anomalię (może być dziesiętna), (musi być większa lub równa 1, domyślnie 3)
- **devalg** (musi być ujęty w podwójny cudzysłów) - algorytm odchylenia; może to być *stddevpop*, *stddevsamp* lub *mad* (domyślnie)
- **s window** - rozpiętość (w opóźnieniach) okna loess dla ekstrakcji sezonowej (domyślnie 10 * liczba wpisów w eval period + 1)

Przykłady:

```
trendstl(/host/key,100h:now/h,10h,2h) #analizuje ostatnie 100 godzin danych trendu, znajduje współczynnik
trendstl(/host/key,100h:now/h-10h,100h,2h,2.1,"mad") #analizuje okres 100 godzin danych trendu do 10 godzi
trendstl(/host/key,100d:now/d-1d,10d,1d,4,,10) #analizuje 100 dni danych trendu do jednego dnia wstecz, zn
trendstl(/host/key,1M:now/M-1y,1d,2h,,"stddevsamp") #analizuje poprzedni miesiąc sprzed roku, znajduje wsp
```

trendsum(/host/key,time period:time shift)

Suma wartości trendów w zdefiniowanym okresie czasu.

Parametry:

- Zobacz wspólne parametry

Przykłady:

```
trendsum(/host/key,1h:now/h) #suma dla poprzedniej godziny (np. 12:00-13:00)
trendsum(/host/key,1h:now/h-1h) #suma sprzed dwóch godzin (11:00-12:00)
trendsum(/host/key,1h:now/h-2h) #suma sprzed trzech godzin (10:00-11:00)
trendsum(/host/key,1M:now/M-1y) #suma dla poprzedniego miesiąca sprzed roku
```

Zobacz wszystkie obsługiwane funkcje.

Funkcje matematyczne

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Funkcje matematyczne są obsługiwane dla typów wartości float i integer, o ile nie zaznaczono inaczej.

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
abs	Wartość bezwzględna wartości.
acos	Arcus cosinus wartości jako kąt wyrażony w radianach.
asin	Arcus sinus wartości jako kąt wyrażony w radianach.
atan	Arcus tangens wartości jako kąt wyrażony w radianach.
atan2	Arcus tangens współrzędnej rzędnej (wartości) i odciętej określonych jako kąt wyrażony w radianach.
avg	Średnia wartość odwołanych wartości pozycji.
cbrt	Pierwiastek sześcienny z wartości.
ceil	Zaokrągla wartość w górę do najbliższej większej lub równej liczby całkowitej.
cos	Cosinus wartości, gdzie wartość jest kątem wyrażonym w radianach.
cosh	Cosinus hiperboliczny wartości.
cot	Cotangens wartości, gdzie wartość jest kątem wyrażonym w radianach.
degrees	Konwertuje wartość z radianów na stopnie.
e	Liczba Eulera (2.718281828459045).
exp	Liczba Eulera podniesiona do potęgi wartości.
expm1	Liczba Eulera podniesiona do potęgi wartości minus 1.
floor	Zaokrągla wartość w dół do najbliższej mniejszej lub równej liczby całkowitej.
log	Logarytm naturalny.
log10	Logarytm dziesiętny.
max	Najwyższa wartość odwołanych wartości pozycji.
min	Najniższa wartość odwołanych wartości pozycji.
mod	Reszta z dzielenia.
pi	Stała Pi (3.14159265358979).
power	Potęga wartości.
radians	Konwertuje wartość ze stopni na radiany.
rand	Zwraca losową wartość całkowitą.
round	Zaokrągla wartość do określonej liczby miejsc po przecinku.
signum	Zwraca '-1', jeśli wartość jest ujemna, '0', jeśli wartość wynosi zero, '1', jeśli wartość jest dodatnia.
sin	Sinus wartości, gdzie wartość jest kątem wyrażonym w radianach.
sinh	Sinus hiperboliczny wartości, gdzie wartość jest kątem wyrażonym w radianach.
sqrt	Pierwiastek kwadratowy z wartości.
sum	Suma odwołanych wartości pozycji.
tan	Tangens wartości.

Function	Description
truncate	Obcina wartość do określonej liczby miejsc po przecinku.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielone przecinkiem
- Wyrażenia są akceptowane jako parametry
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >

abs(value)

Wartość bezwzględna (od 0) wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość bezwzględna zarówno „3”, jak i „-3” będzie wynosić „3”.

Przykład:

```
abs(last(/host/key))>10
```

acos(value)

Arcus cosinus wartości jako kąta, wyrażony w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Wartość musi mieścić się w zakresie od -1 do 1. Na przykład arcus cosinus wartości „0.5” będzie wynosił „2.0943951”.

Przykład:

```
acos(last(/host/key))
```

asin(value)

Arcus sinus wartości jako kąt wyrażony w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Wartość musi mieścić się w zakresie od -1 do 1. Na przykład arcus sinus wartości „0.5” będzie wynosić „-0.523598776”.

Przykład:

```
asin(last(/host/key))
```

atan(value)

Arcus tangens wartości jako kąt wyrażony w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład arcus tangens wartości „1” będzie wynosił „0.785398163”.

Przykład:

```
atan(last(/host/key))
```

atan2(value,abscissa)

Arcus tangens współrzędnych rzędnej (value) i odciętej, określony jako kąt wyrażony w radianach.

Parametry:

- **value** - wartość do sprawdzenia;
- **abscissa** - wartość odciętej.

Na przykład arcus tangens współrzędnych rzędnej i odciętej dla wartości „1” będzie wynosić „2.21429744”.

Przykład:

```
atan2(last(/host/key),2)
```

avg(<value1>,<value2>,...)

Średnia wartość wartości pozycja, do których istnieje odwołanie.

Parametr:

- **valueX** — wartość zwracana przez inną funkcję, która działa na historii pozycji.

Przykład:

```
avg(avg(/host/key,1h),avg(/host2/key2,1h))
```

cbrt(value)

Pierwiastek sześcienny z wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład pierwiastek sześcienny z „64” wyniesie „4”, a z „63” — „3.97905721”.

Przykład:

```
cbrt(last(/host/key))
```

ceil(value)

Zaokrągla wartość w górę do najbliższej większej lub równej liczby całkowitej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład „2.4” zostanie zaokrąglone w górę do „3”. Zobacz także floor().

Przykład:

```
ceil(last(/host/key))
```

cos(value)

Cosinus wartości, gdzie wartość jest kątem wyrażonym w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład cosinus wartości „1” będzie wynosił „0.54030230586”.

Przykład:

```
cos(last(/host/key))
```

cosh(value)

Cosinus hiperboliczny wartości. Zwraca wartość jako liczbę rzeczywistą, a nie w notacji naukowej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład cosinus hiperboliczny wartości „1” wyniesie „1.54308063482”.

Przykład:

```
cosh(last(/host/key))
```

cot(value)

Cotangens wartości, gdzie wartość jest kątem wyrażonym w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład cotangens wartości „1” będzie wynosił „0.54030230586”.

Przykład:

```
cot(last(/host/key))
```

degrees(value)

Konwertuje wartość z radianów na stopnie.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość „1” po konwersji na stopnie będzie wynosić „57.2957795”.

Przykład:

```
degrees(last(/host/key))
```

e

Liczba Eulera (2.718281828459045).

Przykład:

```
e()
```

exp(value)

Liczba Eulera podniesiona do potęgi wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład liczba Eulera podniesiona do potęgi wartości „2” będzie równa „7.38905609893065”.

Przykład:

```
exp(last(/host/key))
```

expm1(value)

Liczba Eulera podniesiona do potęgi wartości minus 1.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład liczba Eulera podniesiona do potęgi wartości „2” minus 1 będzie równa „6.38905609893065”.

Przykład:

```
expm1(last(/host/key))
```

floor(value)

Zaokrągla wartość w dół do najbliższej mniejszej lub równej liczby całkowitej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład „2.6” zostanie zaokrąglone w dół do „2”. Zobacz także ceil().

Przykład:

```
floor(last(/host/key))
```

log(value)

Logarytm naturalny.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład logarytm naturalny wartości „2” wyniesie „0.69314718055994529”.

Przykład:

```
log(last(/host/key))
```

log10(value)

Logarytm dziesiętny.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład logarytm dziesiętny wartości „5” wyniesie „0.69897000433”.

Przykład:

```
log10(last(/host/key))
```

max(<value1>,<value2>,...)

Najwyższa wartość spośród wartości referencyjnych pozycji.

Parametr:

- **valueX** - wartość zwracana przez inną funkcję, która działa na historii pozycji.

Przykład:

```
max(avg(/host/key,1h),avg(/host2/key2,1h))
```

min(<value1>,<value2>,...)

Najniższa wartość spośród wartości referencyjnych pozycji.

Parametr:

- **valueX** - wartość zwracana przez inną funkcję, która działa na historii pozycji.

Przykład:

```
min(avg(/host/key,1h),avg(/host2/key2,1h))
```

mod(value,denominator)

Reszta z dzielenia.

Parametry:

- **value** - wartość do sprawdzenia;
- **denominator** - mianownik dzielenia.

Na przykład reszta z dzielenia wartości ’5’ przez mianownik dzielenia ’2’ będzie wynosić ’1’.

Przykład:

```
mod(last(/host/key),2)
```

pi

Stała Pi (3.14159265358979).

Przykład:

```
pi()
```

power(value,power value)

Potęga wartości.

Parametry:

- **value** - wartość do sprawdzenia;
- **power value** - wykładnik potęgi do użycia.

Na przykład trzecia potęga wartości „2” będzie równa „8”.

Przykład:

```
power(last(/host/key),3)
```

radians(value)

Konwertuje wartość ze stopni na radiany.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość „1” po konwersji na radiany będzie wynosić „0.0174532925”.

Przykład:

```
radians(last(/host/key))
```

rand

Zwraca losową wartość całkowitą. Jest to liczba pseudolosowa generowana z użyciem czasu jako ziarna (wystarczająca do celów matematycznych, ale nie do kryptografii).

Przykład:

```
rand()
```

round(value,decimal places)

Zaokrągla wartość do określonej liczby miejsc dziesiętnych.

Parametry:

- **value** - wartość do sprawdzenia;
- **decimal places** - określa liczbę miejsc dziesiętnych do zaokrąglenia (możliwe jest również 0).

Na przykład wartość „2.5482” zaokrąglona do 2 miejsc dziesiętnych będzie wynosić „2.55”.

Przykład:

```
round(last(/host/key),2)
```

signum(value)

Zwraca ’-1’, jeśli wartość jest ujemna, ’0’, jeśli wartość jest równa zero, oraz ’1’, jeśli wartość jest dodatnia.

Parametr:

- **value** - wartość do sprawdzenia.

Przykład:

```
signum(last(/host/key))
```

sin(value)

Sinus wartości, gdzie wartość jest kątem wyrażonym w radianach.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład sinus wartości „1” wyniesie „0.8414709848”.

Przykład:

```
sin(last(/host/key))
```

sinh(value)

Sinus hiperboliczny wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład sinus hiperboliczny wartości „1” wyniesie „1.17520119364”.

Przykład:

```
sinh(last(/host/key))
```

sqrt(value)

Pierwiastek kwadratowy z wartości.
 Ta funkcja zakończy się niepowodzeniem dla wartości ujemnej.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład pierwiastek kwadratowy z wartości „3.5” będzie wynosił „1.87082869339”.

Przykład:

```
sqrt(last(/host/key))
```

sum(<value1>,<value2>,...)

Suma wartości referencyjnych pozycji.

Parametr:

- **valueX** - wartość zwracana przez inną funkcję, która pracuje z historią pozycji.

Przykład:

```
sum(avg(/host/key,1h),avg(/host2/key2,1h))
```

tan(value)

Tangens wartości.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład tangens wartości „1” wyniesie „1.55740772465”.

Przykład:

```
tan(last(/host/key))
```

truncate(value,decimal places)

Obetnij wartość do określonej liczby miejsc dziesiętnych.

Parametry:

- **value** - wartość do sprawdzenia;
- **decimal places** - określ liczbę miejsc dziesiętnych do obcięcia (możliwe jest również 0).

Na przykład wartość „2.5482” obcięta do 2 miejsc dziesiętnych będzie wynosić „2.54”.

Przykład:

```
truncate(last(/host/key),2)
```

Zobacz wszystkie obsługiwane funkcje.

Funkcje operatorów

Wszystkie wymienione tutaj funkcje są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby wyświetlić pełne szczegóły.

Funkcja	Opis
between	Sprawdza, czy wartość należy do podanego zakresu.
in	Sprawdza, czy wartość jest równa co najmniej jednej z wymienionych wartości.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielone przecinkiem
- Wyrażenia są akceptowane jako parametry

between(value,min,max)

Sprawdza, czy wartość należy do podanego zakresu.
 Obsługiwane typy wartości: *Integer*, *Float*.
 Zwraca: 1 - w zakresie; 0 - w przeciwnym razie.

Parametry:

- **value** - wartość do sprawdzenia;
- **min** - wartość minimalna;
- **max** - wartość maksymalna.

Przykład:

```
between(last(/host/key),1,10)=1 #wyzwalacz, jeśli wartość jest między 1 a 10
```

in(value,value1,value2,...valueN)

Sprawdza, czy wartość jest równa co najmniej jednej z podanych wartości.
 Obsługiwane typy wartości: *Integer*, *Float*, *Character*, *Text*, *Log*.
 Zwraca: 1 - jeśli równa; 0 - w przeciwnym razie.

Parametry:

- **value** - wartość do sprawdzenia;
- **valueX** - podane wartości (wartości tekstowe muszą być ujęte w podwójny cudzysłów).

Wartość jest porównywana z podanymi wartościami jako liczby, jeśli wszystkie te wartości mogą zostać przekonwertowane na wartości liczbowe; w przeciwnym razie jest porównywana jako ciągi znaków.

Przykład:

```
in(last(/host/key),5,10)=1 #wyzwalacz, jeśli ostatnia wartość jest równa 5 lub 10
in("text",last(/host/key),last(/host/key,#2))=1 #wyzwalacz, jeśli "text" jest równe jednej z 2 ostatnich w
```

Zobacz wszystkie obsługiwane funkcje.

Funkcje predykcyjne

Wszystkie funkcje wymienione tutaj są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby zobaczyć pełne szczegóły.

Function	Description
forecast	Przyszła wartość, max, min, delta lub avg pozycji.
timeleft	Czas w sekundach potrzebny, aby pozycja osiągnęła określony próg.

Wspólne parametry

- `/host/key` to wspólny wymagany pierwszy parametr funkcji odwołujących się do historii pozycji hosta.
- `(sec|#num)<:time shift>` to wspólny drugi parametr funkcji odwołujących się do historii pozycji hosta, gdzie:
  - **sec** - maksymalny okres oceny w sekundach (można używać przyrostków czasu), lub
  - **#num** - maksymalny zakres oceny w ostatnio zebranych wartościach (jeśli poprzedzony znakiem hash).
  - **time shift** (opcjonalny) - umożliwia przesunięcie punktu oceny wstecz w czasie. Zobacz więcej informacji na temat określania przesunięcia czasu.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczone przez < >
- Parametry specyficzne dla danej funkcji są opisane przy każdej funkcji
- Parametry `/host/key` oraz `(sec|#num)<:time shift>` nigdy nie mogą być ujmowane w cudzysłów

forecast(/host/key,(sec|#num)<:time shift>,time,<fit>,<mode>)

Przyszła wartość, wartość maksymalna, minimalna, delta lub średnia wartości pozycji.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry;
- **time** - horyzont prognozy w sekundach (np. 1, 60) (można używać przyrostków czasu (np. 30s, 1m)); obsługiwane są wartości ujemne;
- **fit** (opcjonalny; musi być ujęty w podwójny cudzysłów) - funkcja używana do dopasowania danych historycznych. Obsługiwane funkcje dopasowania:
linear - funkcja liniowa (domyślnie)
polynomialN - wielomian stopnia N (1 <= N <= 6)
exponential - funkcja wykładnicza
logarithmic - funkcja logarytmiczna
power - funkcja potęgowa
Należy pamiętać, że *polynomial1* jest równoważne *linear*;
- **mode** (opcjonalny; musi być ujęty w podwójny cudzysłów) - żądany wynik. Obsługiwane tryby:
value - wartość (domyślnie)
max - maksimum
min - minimum
delta - *max-min*
avg - średnia
Należy pamiętać, że *value* szacuje wartość pozycji w chwili `now + time`; tryby *max*, *min*, *delta* i *avg* analizują oszacowanie wartości pozycji w przedziale od `now` do `now + time`.

Uwagi:

- Jeśli zwracana wartość jest większa niż 1.7976931348623158E+308 lub mniejsza niż -1.7976931348623158E+308, zostaje odpowiednio ograniczona do 1.7976931348623158E+308 lub -1.7976931348623158E+308;
- Staje się nieobsługiwana tylko w przypadku nieprawidłowego użycia w wyrażeniu (nieprawidłowy typ pozycji, nieprawidłowe parametry); w przeciwnym razie w przypadku błędów zwraca -1;
- Zobacz także dodatkowe informacje na temat predykcyjnych funkcji wyzwalaczy.

Przykłady:

```
forecast(/host/key,#10,1h) #prognoza wartości pozycji za godzinę na podstawie 10 ostatnich wartości
forecast(/host/key,1h,30m) #prognoza wartości pozycji za 30 minut na podstawie danych z ostatniej godziny
forecast(/host/key,1h:now-1d,12h) #prognoza wartości pozycji za 12 godzin na podstawie danych z jednej god
forecast(/host/key,1h,10m,"exponential") #prognoza wartości pozycji za 10 minut na podstawie danych z osta
forecast(/host/key,1h,2h,"polynomial3","max") #prognoza maksymalnej wartości, jaką pozycja może osiągnąć w
forecast(/host/key,#2,-20m) #oszacowanie wartości pozycji sprzed 20 minut na podstawie dwóch ostatnich war
```

timeleft(/host/key,(sec|#num)<:przesunięcie czasu>,threshold,<fit>)

Czas w sekundach potrzebny, aby pozycja osiągnęła określony próg.
 Obsługiwane typy wartości: *Float*, *Integer*.

Parametry:

- Zobacz wspólne parametry;
- **threshold** - wartość do osiągnięcia (można używać przyrostków jednostek);
- **fit** (opcjonalny; musi być ujęty w podwójny cudzysłów) - zobacz forecast().

Uwagi:

- Jeśli wartość do zwrócenia jest większa niż 1.7976931348623158E+308, wartość zwracana zostanie ograniczona do 1.7976931348623158E+308;
- Zwraca 1.7976931348623158E+308, jeśli nie można osiągnąć progu;
- Staje się nieobsługiwana tylko w przypadku nieprawidłowego użycia w wyrażeniu (nieprawidłowy typ pozycji, nieprawidłowe parametry); w przeciwnym razie w przypadku błędów zwraca -1;
- Zobacz także dodatkowe informacje na temat predykcyjnych funkcji wyzwalaczy.

Przykłady:

```
timeleft(/host/key,#10,0) #czas do osiągnięcia przez wartość pozycji zera na podstawie ostatnich 10 wartoś
timeleft(/host/key,1h,100) #czas do osiągnięcia przez wartość pozycji 100 na podstawie danych z ostatniej
timeleft(/host/key,1h:now-1d,100) #czas do osiągnięcia przez wartość pozycji 100 na podstawie jednej godzi
timeleft(/host/key,1h,200,"polynomial2") #czas do osiągnięcia przez wartość pozycji 200 na podstawie danyc
```

Zobacz wszystkie obsługiwane funkcje.

Predykcyjne funkcje wyzwalaczy

Przegląd

Czasami pojawiają się oznaki nadchodzącego problemu. Oznaki te można wykryć, aby z wyprzedzeniem podjąć działania zapobiegające problemowi lub przynajmniej zminimalizować jego wpływ.

Zabbix udostępnia narzędzia do przewidywania przyszłego zachowania monitorowanego systemu na podstawie danych historycznych. Narzędzia te są realizowane za pomocą predykcyjnych funkcji wyzwalacza.

Funkcje

Przed ustawieniem wyzwalacza należy określić, czym jest stan problemowy i ile czasu potrzeba na podjęcie działania. Następnie są dwa sposoby skonfigurowania wyzwalacza sygnalizującego potencjalną niepożądaną sytuację. Po pierwsze: wyzwalacz musi zostać uruchomiony, gdy oczekuje się, że system będzie w stanie problemowym po upływie „czasu na działanie". Po drugie: wyzwalacz musi zostać uruchomiony, gdy system osiągnie stan problemowy w czasie krótszym niż „czas na działanie". Odpowiadające funkcje wyzwalacza, których należy użyć, to **forecast** oraz **timeleft**. Należy pamiętać, że podstawowa analiza statystyczna jest zasadniczo identyczna dla obu funkcji. Możesz skonfigurować wyzwalacz w dowolny preferowany sposób, uzyskując podobne wyniki.

Parametry

Obie funkcje używają niemal tego samego zestawu parametrów. Aby uzyskać informacje, skorzystaj z listy obsługiwanych funkcji.

Przedział czasu

Przede wszystkim należy określić historyczny okres, który Zabbix ma analizować, aby przygotować prognozę. Robi się to w znany sposób za pomocą parametru `time period` oraz opcjonalnego przesunięcia czasu, tak jak w przypadku funkcji **avg**, **count**, **delta**, **max**, **min** i **sum**.

Horyzont prognozowania

(tylko **forecast**)
 Parametr `time` określa, jak daleko w przyszłość Zabbix powinien ekstrapolować zależności znalezione w danych historycznych. Niezależnie od tego, czy używasz `time_shift`, czy nie, `time` jest zawsze liczony od bieżącego momentu.

Próg do osiągnięcia

(tylko **timeleft**)
 Parametr `threshold` określa wartość, którą analizowana pozycja ma osiągnąć, bez względu na to, czy od góry, czy od dołu. Gdy wyznaczymy f(t) (patrz poniżej), należy rozwiązać równanie f(t) = `threshold` i zwrócić pierwiastek, który jest bliżej chwili obecnej i znajduje się na prawo od niej, lub 1.7976931348623158E+308, jeśli taki pierwiastek nie istnieje.

Note:
Gdy wartości pozycji zbliżają się do progu, a następnie go przekraczają, **timeleft** zakłada, że punkt przecięcia znajduje się już w przeszłości, i dlatego przełącza się na następne przecięcie z poziomem `threshold`, jeśli takie istnieje. Zalecaną praktyką jest używanie prognoz jako uzupełnienia zwykłej diagnostyki problemów, a nie jako jej zamiennika.[1]

Funkcje dopasowania

Domyślną funkcją `fit` jest funkcja *liniowa*. Jeśli jednak monitorowany system jest bardziej złożony, masz do wyboru więcej opcji.

`fit`	x = f(t)
linear	$x = a + b*t$
polynomialN[2]	$x = a_0 + a_1*t + a_2*t^2 + ... + a_n*t^n$
exponential	$x = a*exp(b*t)$
logarithmic	$x = a + b*log(t)$
power	$x = a*t^b$

Tryby

(tylko **forecast**)
 Za każdym razem, gdy funkcja wyzwalacza jest obliczana, pobiera dane z określonego okresu historii i dopasowuje do nich wskazaną funkcję. Tak więc, jeśli dane są nieco inne, dopasowana funkcja również będzie nieco inna. Jeśli po prostu obliczymy wartość dopasowanej funkcji w określonym momencie w przyszłości, nie będziesz wiedzieć nic o tym, jak analizowana pozycja ma się zachowywać od teraz do tego momentu w przyszłości. Dla niektórych opcji `fit` (takich jak *polynomial*) prosta wartość z przyszłości może być myląca.

`mode`	wynik **forecast**
value	f(teraz + `time`)
max	$max_{teraz <= t <= teraz + time}$ f(t)
min	$min_{teraz <= t <= teraz + time}$ f(t)
delta	*max - min*
avg	średnia z f(t) (teraz <= t <= teraz + `time`) zgodnie z definicją

Szczegóły

Aby uniknąć obliczeń na bardzo dużych liczbach, traktujemy znacznik czasu pierwszej wartości w określonym okresie plus 1 ns jako nowy czas zerowy (aktualny czas epoki jest rzędu 10^9, kwadrat epoki to 10^{18}, precyzja podwójna wynosi około 10^{-16}). 1 ns dodaje się, aby zapewnić wyłącznie dodatnie wartości czasu dla dopasowań *logarytmicznych* i *potęgowych*, które obejmują obliczanie log(t). Przesunięcie czasu nie wpływa na funkcje *liniowe*, *wielomianowe*, *wykładnicze* (poza łatwiejszymi i dokładniejszymi obliczeniami), ale zmienia kształt funkcji *logarytmicznych* i *potęgowych*.

Potencjalne błędy

Funkcje zwracają -1 w następujących sytuacjach:

- określony okres oceny nie zawiera danych;
- wynik operacji matematycznej nie jest zdefiniowany[3];
- komplikacje numeryczne (niestety, dla niektórych zestawów danych wejściowych zakres i precyzja formatu liczb zmiennoprzecinkowych podwójnej precyzji okazują się niewystarczające)[4].

Note:
Nie są zgłaszane żadne ostrzeżenia ani błędy, jeśli wybrane dopasowanie słabo opisuje dostarczone dane lub jeśli danych jest po prostu zbyt mało do dokładnego przewidywania.

Przykłady i obsługa błędów

Aby otrzymać ostrzeżenie, gdy na host zaczyna brakować wolnego miejsca na dysku, możesz użyć wyrażenia wyzwalacza takiego jak to:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h
```

Jednak kod błędu -1 może wejść w grę i wprowadzić wyzwalacz w stan problemu. Ogólnie jest to dobre, ponieważ otrzymujesz ostrzeżenie, że Twoje prognozy nie działają poprawnie i powinieneś przyjrzeć się im dokładniej, aby ustalić przyczynę. Czasami jednak jest to niepożądane, ponieważ -1 może po prostu oznaczać, że w ciągu ostatniej godziny nie uzyskano danych o wolnym miejscu na dysku hosta. Jeśli otrzymujesz zbyt wiele fałszywie dodatnich alertów, rozważ użycie bardziej złożonego wyrażenia wyzwalacza [5]:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1
```

Sytuacja jest nieco trudniejsza w przypadku **forecast**. Przede wszystkim -1 może, ale nie musi, wprowadzić wyzwalacz w stan problemu w zależności od tego, czy masz wyrażenie takie jak `forecast(/host/item,(...))<...`, czy takie jak `forecast(/host/item,(...))>...`

Ponadto -1 może być prawidłową prognozą, jeśli ujemna wartość pozycji jest czymś normalnym. Jednak prawdopodobieństwo takiej sytuacji w rzeczywistych warunkach jest znikome (zobacz, jak działa operator =). Dlatego dodaj `... or forecast(/host/item,(...))=-` lub `... and forecast(/host/item,(...))<>-1`, jeśli odpowiednio chcesz lub nie chcesz traktować -1 jako problemu.

Przypisy

[1] Na przykład prosty wyzwalacz taki jak `timeleft(/host/item,1h,X) < 1h` może przejść w stan problemu, gdy wartość pozycji zbliża się do X, a następnie nagle powrócić do stanu OK po osiągnięciu wartości X. Jeśli problem polega na tym, że wartość pozycji jest poniżej X, użyj: `last(/host/item) < X or timeleft(/host/item,1h,X) < 1h` Jeśli problem polega na tym, że wartość pozycji jest powyżej X, użyj: `last(/host/item) > X or timeleft(/host/item,1h,X) < 1h`

[2] Stopień wielomianu może wynosić od 1 do 6, *polynomial1* jest równoważny *linear*. Jednak wielomianów wyższego stopnia należy używać ostrożnie. Jeśli okres oceny zawiera mniej punktów niż potrzeba do wyznaczenia współczynników wielomianu, stopień wielomianu zostanie obniżony (np. żądany jest *polynomial5*, ale są tylko 4 punkty, dlatego zostanie dopasowany *polynomial3*).

[3] Na przykład dopasowanie funkcji *exponential* lub *power* obejmuje obliczanie log() wartości pozycji. Jeśli dane zawierają zera lub liczby ujemne, otrzymasz błąd, ponieważ log() jest zdefiniowany tylko dla wartości dodatnich.

[4] Dla dopasowań *linear*, *exponential*, *logarithmic* i *power* wszystkie niezbędne obliczenia można zapisać jawnie. Dla *polynomial* bez dodatkowych kroków można obliczyć tylko *value*. Obliczenie *avg* obejmuje wyznaczenie funkcji pierwotnej wielomianu (analitycznie). Obliczenie *max*, *min* i *delta* obejmuje wyznaczenie pochodnej wielomianu (analitycznie) i znalezienie jej pierwiastków (numerycznie). Rozwiązanie f(t) = 0 obejmuje znalezienie pierwiastków wielomianu (numerycznie).

[5] Jednak w tym przypadku -1 może spowodować, że wyzwalacz wyjdzie ze stanu problemu. Aby zapewnić pełną ochronę, użyj: `timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and ({TRIGGER.VALUE}=0 and timeleft(/host/vfs.fs.size[/,free], or {TRIGGER.VALUE}=1)`

Funkcje ciągów znaków

Wszystkie wymienione tutaj funkcje są obsługiwane w:

- Wyrażeniach wyzwalaczy
- Formułach obliczanych pozycji
- Makrach wyrażeń

Funkcje są wymienione bez dodatkowych informacji. Kliknij funkcję, aby wyświetlić pełne szczegóły.

Funkcja	Opis
ascii	Kod ASCII skrajnego lewego znaku wartości.
bitlength	Długość wartości w bitach.
bytelength	Długość wartości w bajtach.
char	Zwraca znak, interpretując wartość jako kod ASCII.
concat	Ciąg znaków powstały przez połączenie wartości wskazanych pozycji lub wartości stałych.
contains	Zwraca 1, jeśli wartość zawiera podciąg określony przez `search value`, w przeciwnym razie 0.
insert	Wstawia określone znaki lub spacje do ciągu znaków, zaczynając od określonej pozycji w ciągu.

Funkcja	Opis
jsonpath	Zwraca wynik JSONPath.
left	Zwraca skrajne lewe znaki wartości.
length	Długość wartości w znakach.
ltrim	Usuwa określone znaki z początku ciągu.
mid	Zwraca podciąg o długości N znaków, zaczynający się od pozycji znaku określonej przez „start”. Ta funkcja jest aliasem funkcji `substring()`.
repeat	Powtarza ciąg znaków.
replace	Znajduje wzorzec w wartości i zastępuje go określonym tekstem.
right	Zwraca skrajne prawe znaki wartości.
rtrim	Usuwa określone znaki z końca ciągu.
substring	Zwraca podciąg o długości N znaków, zaczynający się od pozycji znaku określonej przez „start”.
trim	Usuwa określone znaki z początku i końca ciągu.
xmlxpath	Zwraca wynik XML XPath.

Szczegóły funkcji

Kilka ogólnych uwag dotyczących parametrów funkcji:

- Parametry funkcji są oddzielane przecinkiem
- Wyrażenia są akceptowane jako parametry
- Parametry tekstowe muszą być ujęte w podwójny cudzysłów; w przeciwnym razie mogą zostać błędnie zinterpretowane
- Opcjonalne parametry funkcji (lub części parametrów) są oznaczane przez < >

ascii(value)

Kod ASCII skrajnie lewego znaku wartości.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametr:

- **value** - wartość do sprawdzenia

Na przykład wartość taka jak ’Abc’ zwróci ’65’ (kod ASCII dla ’A’).

Przykład:

```
ascii(last(/host/key))
```

bitlength(value)

Długość wartości w bitach.
 Obsługiwane typy wartości: *String*, *Text*, *Log*, *Integer*.

Parametr:

- **value** - wartość do sprawdzenia

Przykład:

```
bitlength(last(/host/key))
```

bytelength(value)

Długość wartości w bajtach.
 Obsługiwane typy wartości: *String*, *Text*, *Log*, *Integer*.

Parametr:

- **value** - wartość do sprawdzenia

Przykład:

```
bytelength(last(/host/key))
```

char(value)

Zwraca znak poprzez interpretację wartości jako kodu ASCII.
 Obsługiwane typy wartości: *Integer*.

Parametr:

- **value** — wartość do sprawdzenia

Wartość musi mieścić się w zakresie 0–255. Na przykład wartość taka jak „65” (interpretowana jako kod ASCII) zwróci „A”.

Przykład:

```
char(last(/host/key))
```

concat(<value1>,<value2>,...)

Ciąg znaków wynikający z połączenia wartości referencyjnych pozycji lub wartości stałych.
 Obsługiwane typy wartości: *String*, *Text*, *Log*, *Float*, *Integer*.

Parametr:

- **valueX** - wartość zwracana przez jedną z funkcji historii lub wartość stała (ciąg znaków, liczba całkowita lub zmienno-przecinkowa). Musi zawierać co najmniej dwa parametry.

Na przykład wartość taka jak 'Zab' połączona z 'bix' (stały ciąg znaków) zwróci 'Zabbix'.

Przykłady:

```
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)

Zwraca 1, jeśli wartość zawiera podciąg określony przez `search value`, w przeciwnym razie zwraca 0.| Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **search value** - wartość do wyszukania.

Uwagi:

- Wyszukiwanie uwzględnia wielkość liter.
- Jeśli wartość jest prawidłowa, a `search value` jest pusty, funkcja zwraca 1.

Na przykład funkcja zwraca 1, jeśli wartość zawiera „Zabbix".

Przykład:

```
contains(last(/host/key),"Zabbix")
```

insert(value,start,length,replacement)

Wstawia określone znaki lub spacje do ciągu znaków, zaczynając od wskazanej pozycji w ciągu.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **start** - pozycja początkowa;
- **length** - pozycje do zastąpienia;
- **replacement** - ciąg zastępujący.

Na przykład wartość taka jak 'Zabbbix' zostanie zastąpiona przez 'Zabbix', jeśli 'bb' (pozycja początkowa 3, liczba pozycji do zastąpienia 2) zostanie zastąpione przez 'b'.

Przykład:

```
insert(last(/host/key),3,2,"b")
```

jsonpath(value,path,<default>)

Zwraca wynik JSONPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **path** - ścieżka (musi być ujęta w cudzysłów);
- **default** - opcjonalna wartość zapasowa, jeśli zapytanie JSONPath nie zwróci danych. Należy pamiętać, że w przypadku innych błędów zwracane jest niepowodzenie (np. „unsupported construct").

Przykład:

```
jsonpath(last(/host/proc.get[zabbix_agentd,,,summary]),"$..size")
```

left(value,count)

Zwraca skrajnie lewe znaki wartości.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;

- **count** - liczba znaków do zwrócenia.

Na przykład można zwrócić „Zab” z „Zabbix”, określając 3 skrajnie lewe znaki do zwrócenia. Zobacz także right().

Przykład:

```
left(last(/host/key),3) #zwraca trzy skrajnie lewe znaki
```

length(value)

Długość wartości w znakach.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametr:

- **value** - wartość do sprawdzenia.

Przykłady:

```
length(last(/host/key)) #długość najnowszej wartości
length(last(/host/key,#3)) #długość trzeciej najnowszej wartości
length(last(/host/key,#1:now-1d)) #długość najnowszej wartości sprzed jednego dnia
```

ltrim(value,<chars>)

Usuń określone znaki z początku ciągu.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **chars** (opcjonalnie) - określa znaki do usunięcia.

Domyślnie usuwane są białe znaki z początku ciągu (jeśli nie określono opcjonalnych znaków). Zobacz także: rtrim(), trim().

Przykłady:

```
ltrim(last(/host/key)) #usuń białe znaki z początku ciągu
ltrim(last(/host/key),"Z") #usuń dowolne 'Z' z początku ciągu
ltrim(last(/host/key)," Z") #usuń dowolną spację i 'Z' z początku ciągu
```

mid(value,start,length)

Zwraca podciąg o długości N znaków, rozpoczynający się na pozycji znaku określonej przez parametr „start”.
Ta funkcja jest aliasem funkcji `substring()`.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **start** - pozycja początkowa podciągu;
- **length** - liczba pozycji do zwrócenia w podciągu.

Na przykład można zwrócić „abbi” z wartości „Zabbix”, jeśli pozycja początkowa wynosi 2, a liczba pozycji do zwrócenia wynosi 4.

Przykład:

```
mid(last(/host/key),2,4)="abbi"
```

repeat(value,count)

Powtórz ciąg znaków.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **count** - liczba powtórzeń.

Przykład:

```
repeat(last(/host/key),2) #powtórz wartość dwa razy
```

replace(value,pattern,replacement)

Znajduje wzorzec w wartości i zastępuje go ciągiem replacement. Wszystkie wystąpienia wzorca zostaną zastąpione.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **pattern** - wzorzec do znalezienia;
- **replacement** - ciąg, którym zostanie zastąpiony wzorzec.

Przykład:

```
replace(last(/host/key),"ibb","abb") #zastępuje wszystkie 'ibb' przez 'abb'
```

right(value,count)

Zwraca skrajnie prawe znaki wartości.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **count** - liczba znaków do zwrócenia.

Na przykład można zwrócić „bix” z „Zabbix”, podając 3 skrajnie prawe znaki do zwrócenia. Zobacz także left().

Przykład:

```
right(last(/host/key),3) #zwraca trzy skrajnie prawe znaki
```

rtrim(value,<chars>)

Usuwa określone znaki z końca ciągu znaków.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **chars** (opcjonalnie) - określa znaki do usunięcia.

Domyślnie usuwane są białe znaki z prawej strony (jeśli nie określono opcjonalnych znaków). Zobacz także: ltrim(), trim().

Przykłady:

```
rtrim(last(/host/key)) #usuń białe znaki z końca ciągu znaków
rtrim(last(/host/key),"x") #usuń wszystkie znaki 'x' z końca ciągu znaków
rtrim(last(/host/key),"x ") #usuń wszystkie znaki 'x' i spacje z końca ciągu znaków
```

substring(value,start,length)

Zwraca podciąg o długości N znaków, rozpoczynający się na pozycji znaku określonej przez parametr „start”.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **start** - pozycja początkowa podciągu;
- **length** - liczba pozycji zwracanych w podciągu.

Na przykład można zwrócić „abbi” z wartości „Zabbix”, jeśli pozycja początkowa wynosi 2, a liczba zwracanych pozycji wynosi 4.

Przykład:

```
substring(last(/host/key),2,4)="abbi"
```

trim(value,<chars>)

Usuń określone znaki z początku i końca ciągu znaków.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **chars** (opcjonalnie) - określa znaki do usunięcia.

Domyślnie z obu stron usuwane są białe znaki (jeśli nie określono opcjonalnych znaków). Zobacz także: ltrim(), rtrim().

Przykłady:

```
trim(last(/host/key)) #usuń białe znaki z początku i końca ciągu znaków
trim(last(/host/key),"_") #usuń '_' z początku i końca ciągu znaków
```

xmlxpath(value,path,<default>)

Zwraca wynik XML XPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **value** - wartość do sprawdzenia;
- **path** - ścieżka (musi być ujęta w cudzysłów);

- **default** - opcjonalna wartość zastępcza, jeśli zapytanie XML XPath zwróci pusty zbiór węzłów. Nie zostanie zwrócona, jeśli pusty wynik nie jest zbiorem węzłów (tj. pustym ciągiem). W przypadku innych błędów zwracane jest niepowodzenie (np. „invalid expression").

Przykład:

```
xmlxpath(last(/host/xml_result),"/response/error/status")
```

Zobacz wszystkie obsługiwane funkcje.

Najlepsze praktyki dotyczące wyzwalaczy

Wyzwalacze są potężnym narzędziem, ale mogą też generować niepożądany szum alertów. Aby zobaczyć więcej rzeczywistych sygnałów i mniej szumu, zastosuj się do poniższych wskazówek:

1. Zmniejsz czułość wyzwalaczy. Zamiast alertować na podstawie najnowszej wartości (zbyt wysoka/zbyt niska), analizuj średnią z dłuższego okresu, używając funkcji takich jak **avg**, **min** i **max**.
2. Rozważ użycie funkcji **percentile** (ustaw ją na 95% lub 5%) w wyzwalaczach, jeśli chcesz uniknąć alertów wywoływanych przez losowe skoki i spadki.
3. Użyj histerezy, aby uniknąć „flappingu" wyzwalaczy — częstych zmian stanu wyzwalacza (OK ↔ Problem). Ciągłość dla stanu problemu można zdefiniować, dodając wyrażenie odzyskiwania (oddzielny warunek rozwiązania problemu).
4. Użyj zależności wyzwalaczy, aby uniknąć alertów niezwiązanych z przyczyną źródłową.
5. Użyj rangi ważności wyzwalaczy, aby alertować tylko o poważniejszych problemach.
6. Zdefiniuj okna maintenance.

Histereza

Czasami potrzebny jest przedział między stanem problemu a stanem odzyskania, zamiast prostego progu. Na przykład, jeśli chcemy zdefiniować wyzwalacz, który zgłasza problem, gdy temperatura w serwerowni przekroczy 20°C, i chcemy, aby pozostawał on w stanie problemu, dopóki temperatura nie spadnie poniżej 15°C, prosty próg wyzwalacza ustawiony na 20°C nie będzie wystarczający.

Zamiast tego najpierw musimy zdefiniować wyrażenie wyzwalacza dla zdarzenia problemowego (temperatura powyżej 20°C). Następnie musimy zdefiniować dodatkowy warunek odzyskania (temperatura poniżej 15°C). Robi się to przez zdefiniowanie *wyrażenia odzyskania* podczas konfigurowania wyzwalacza.

W tym przypadku odzyskanie po problemie odbywa się w dwóch krokach:

- Najpierw wyrażenie problemowe (temperatura powyżej 20°C) musi zostać obliczone jako FALSE
- Następnie wyrażenie odzyskania (temperatura poniżej 15°C) musi zostać obliczone jako TRUE

Wyrażenie odzyskania jest obliczane dopiero po rozwiązaniu zdarzenia problemowego. Samo ustawienie wyrażenia odzyskania na TRUE nie powoduje rozwiązania problemu, jeśli wyrażenie problemowe nadal ma wartość TRUE!

Przykład

Temperatura w serwerowni jest zbyt wysoka.

Wyrażenie problemu:

```
last(/server/temp)>20
```

Wyrażenie odzyskiwania:

```
last(/server/temp)<=15
```

> **Note:**
> Używanie makra {TRIGGER.VALUE} w wyrażeniu odzyskiwania jest nieefektywne, ponieważ to wyrażenie jest oceniane tylko wtedy, gdy wyzwalacz znajduje się w stanie „Problem". W konsekwencji podczas obliczania wyrażenia makro {TRIGGER.VALUE} zawsze zostanie rozwinięte do wartości „1" (co oznacza stan „Problem").

Zależności wyzwalaczy

Zależności wyzwalaczy mogą być używane do unikania alertów, które nie są związane z główną przyczyną.

Zobacz wszystkie najlepsze praktyki.

Przegląd

Czasami dostępność jednego hosta zależy od innego. Serwer znajdujący się za routerem stanie się niedostępny, jeśli router przestanie działać. Przy wyzwalaczach skonfigurowanych dla obu hostów możesz otrzymać powiadomienia o awarii dwóch hostów, podczas gdy winowajcą był tylko router.

W takiej sytuacji przydatna może być zależność między hostami. Po skonfigurowaniu zależności powiadomienia dotyczące elementów zależnych mogą zostać wstrzymane i wysłane może zostać tylko powiadomienie o problemie głównym.

Chociaż Zabbix nie obsługuje bezpośrednio zależności między hostami, można je definiować za pomocą innej, bardziej elastycznej metody - zależności wyzwalaczy. Wyzwalacz może zależeć od jednego lub większej liczby innych wyzwalaczy.

W naszym prostym przykładzie otwieramy formularz konfiguracji wyzwalacza serwera i ustawiamy jego zależność od odpowiedniego wyzwalacza routera. Przy takiej zależności wyzwalacz serwera nie zmieni swojego stanu, dopóki wyzwalacz, od którego zależy, będzie w stanie „PROBLEM" - w związku z tym nie zostaną wykonane żadne akcje zależne ani wysłane żadne powiadomienia.

Jeśli zarówno serwer, jak i router są niedostępne oraz skonfigurowano zależność między wyzwalaczem „serwer niedostępny" a wyzwalaczem „router niedostępny", Zabbix nie wykona akcji dla wyzwalacza zależnego.

Gdy wyzwalacz nadrzędny jest w stanie PROBLEM, jego elementy zależne mogą zgłaszać wartości, którym nie można ufać. Dlatego wyzwalacze zależne nie będą ponownie oceniane, dopóki wyzwalacz nadrzędny (router w powyższym przykładzie):

- nie powróci ze stanu „PROBLEM" do stanu „OK";
- nie zmieni stanu z „PROBLEM" na „UNKNOWN";
- nie zostanie zamknięty ręcznie, przez korelację lub za pomocą funkcji date and time i/lub nodata();
- nie zostanie rozwiązany przez wartość pozycji, która nie jest używana w wyzwalaczu zależnym;
- nie zostanie wyłączony, nie będzie miał wyłączonej pozycji ani hosta wyłączonej pozycji

We wszystkich wymienionych powyżej przypadkach wyzwalacz zależny (serwer) zostanie ponownie oceniony dopiero po otrzymaniu nowej metryki dla tego wyzwalacza. Oznacza to, że wyzwalacz zależny może nie zostać zaktualizowany natychmiast.

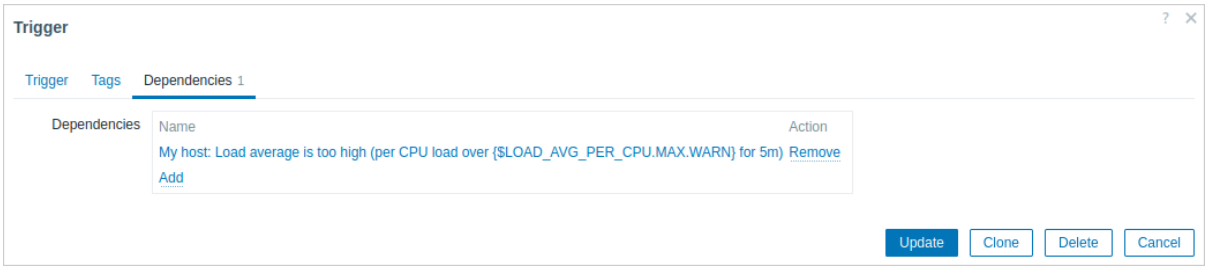
Ponadto:

- Zależność wyzwalacza można dodać od wyzwalacza dowolnego hosta do dowolnego innego wyzwalacza hosta, o ile nie spowoduje to powstania zależności cyklicznej.
- Zależność wyzwalacza można dodać z jednego szablonu do innego. Jeśli wyzwalacz z szablonu A zależy od wyzwalacza z szablonu B, szablon A można połączyć z hostem (lub innym szablonem) tylko razem z szablonem B, natomiast szablon B można połączyć z hostem (lub innym szablonem) samodzielnie.
- Zależność wyzwalacza można dodać od wyzwalacza szablonu do wyzwalacza hosta. W takim przypadku połączenie takiego szablonu z hostem utworzy wyzwalacz hosta zależny od tego samego wyzwalacza szablonu, od którego zależał wyzwalacz. Umożliwia to na przykład utworzenie szablonu, w którym niektóre wyzwalacze zależą od wyzwalaczy routera (hosta). Wszystkie hosty połączone z tym szablonem będą zależeć od konkretnego routera.
- Nie można dodać zależności wyzwalacza od wyzwalacza hosta do wyzwalacza szablonu.
- Zależność wyzwalacza można dodać od prototypu wyzwalacza do innego prototypu wyzwalacza (w ramach tej samej reguły wykrywania niskopoziomowego) lub do rzeczywistego wyzwalacza. Prototyp wyzwalacza nie może zależeć od prototypu wyzwalacza z innej reguły LLD ani od wyzwalacza utworzonego na podstawie prototypu wyzwalacza. Prototyp wyzwalacza hosta nie może zależeć od wyzwalacza z szablonu.

Konfiguracja

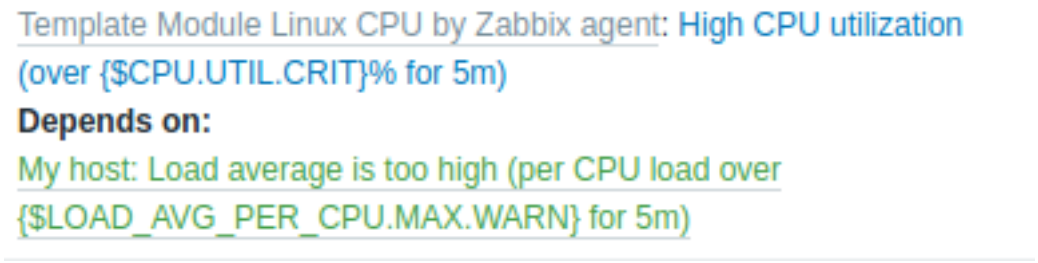
Aby zdefiniować zależność:

1. Otwórz kartę *Zależności* w formularzu konfiguracji wyzwalacza.
2. Kliknij *Dodaj* w sekcji *Zależności* i wybierz co najmniej jeden wyzwalacz, od którego będzie zależeć dany wyzwalacz.



3. Kliknij *Aktualizuj*.

Teraz na liście wyzwalacz będzie oznaczony jako zależny.



Przykład kilku zależności

Na przykład host znajduje się za Router2, a Router2 znajduje się za Router1.

```
Zabbix - Router1 - Router2 - Host
```

Jeśli Router1 nie działa, to oczywiście host i Router2 są również niedostępne, jednak otrzymywanie trzech powiadomień o tym, że host, Router1 i Router2 nie działają, jest przesadą.

W takim przypadku definiujemy dwie zależności:

```
wyzwalacz 'Host is down' zależy od wyzwalacza 'Router2 is down'
wyzwalacz 'Router2 is down' zależy od wyzwalacza 'Router1 is down'
```

Przed zmianą stanu wyzwalacza 'Host is down' Zabbix sprawdzi odpowiednie zależności wyzwalaczy. Jeśli takie zostaną znalezione, a jeden z tych wyzwalaczy znajduje się w stanie **Problem**, stan wyzwalacza nie zostanie zmieniony, akcje nie zostaną wykonane i nie zostaną wysłane żadne powiadomienia.

Zabbix wykonuje to sprawdzenie rekurencyjnie. Jeśli Router1 lub Router2 są niedostępne, wyzwalacz hosta nie zostanie zaktualizowany.

Ważność wyzwalacza

Ważność wyzwalacza może być używana do ograniczenia szumu alertów w przypadku mniej istotnych problemów.

Zobacz wszystkie najlepsze praktyki.

Przegląd

Ważność określa względne znaczenie wyzwalacza.



Przypisanie ważności jest przydatne nie tylko do reprezentacji wizualnej, ale także do unikania powiadomień o błahych problemach.

Ważności wyzwalaczy są używane do:

- Reprezentacji wizualnej (odrębny kolor dla każdego utworzonego problemu, na podstawie oryginalnej ważności wyzwalacza)
- Ograniczania powiadomień na podstawie ważności (zobacz warunki akcji)
- Mediów użytkownika — różne media (kanały powiadomień) dla różnych ważności (na przykład SMS dla wyzwalaczy o ważności *Wysoka* i *Katastrofa* oraz e-mail dla wyzwalaczy o innych ważnościach)
- Dźwięku w alarmach globalnych — różne dźwięki dla różnych ważności

Domyślnie obsługiwane są następujące ważności wyzwalaczy:

Ważność	Kolor	Opis
Niesklasyfikowane	Szary	Może być używana tam, gdzie poziom ważności zdarzenia jest nieznany, nie został określony, nie należy do regularnego zakresu monitorowania itp., na przykład podczas początkowej konfiguracji, jako symbol zastępczy do przyszłej oceny lub jako część procesu integracji.
Informacja	Jasnoniebieski	Może być używana dla zdarzeń informacyjnych, które nie wymagają natychmiastowej uwagi, ale nadal mogą dostarczać cennych informacji.
Ostrzeżenie	Żółty	Może być używana do wskazania potencjalnego problemu, który może wymagać zbadania lub działania, ale nie jest krytyczny.
Średnia	Pomarańczowy	Może być używana do wskazania istotnego problemu, którym należy zająć się stosunkowo szybko, aby zapobiec dalszym problemom.
Wysoka	Jasnoczerwony	Może być używana do wskazania krytycznych problemów, które wymagają natychmiastowej uwagi, aby uniknąć znaczących zakłóceń.
Katastrofa	Czerwony	Może być używana do wskazania poważnego incydentu, który wymaga natychmiastowego działania, aby zapobiec na przykład awariom systemu lub utracie danych.

Nazwy i kolory ważności wyzwalaczy można **dostosować** w *Administracja* > *Ogólne* > *Opcje wyświetlania wyzwalaczy*.
Kolory są współdzielone przez wszystkie motywy frontend.

Tłumaczenie niestandardowych nazw poziomów ważności

> **Attention:**
> Jeśli używane są tłumaczenia frontend Zabbix, niestandardowe nazwy poziomów ważności domyślnie zastąpią nazwy przetłumaczone.

Domyślne nazwy poziomów ważności wyzwalaczy są dostępne do tłumaczenia we wszystkich ustawieniach regionalnych.
Jeśli nazwa poziomu ważności zostanie zmieniona, nazwa niestandardowa będzie używana we wszystkich ustawieniach regionalnych i konieczne będzie dodatkowe ręczne tłumaczenie.

Procedura tłumaczenia niestandardowej nazwy poziomu ważności:

- ustaw wymaganą niestandardową nazwę poziomu ważności, na przykład „Important”
- edytuj <frontend_dir>/locale/<required_locale>/LC_MESSAGES/frontend.po
- dodaj 2 linie:

```
msgid "Important"
msgstr "<ciąg tłumaczenia>"
```

i zapisz plik.

- utwórz pliki .mo zgodnie z opisem w <frontend_dir>/locale/README

Tutaj **msgid** powinno odpowiadać nowej niestandardowej nazwie poziomu ważności, a **msgstr** powinno być jej tłumaczeniem w danym języku.

Tę procedurę należy wykonać po każdej zmianie nazwy poziomu ważności.

Wizualizacja

Przegląd

Wizualizacja zapewnia różne sposoby wyświetlania danych monitoringu zbieranych przez Zabbix. Wykresy pokazują trendy i metryki, mapy sieci wyświetlają topologię infrastruktury, a pulpity łączą konfigurowalne widgety, aby zapewnić kompleksowe widoki monitoringu.

Narzędzia wizualizacji

- Wykresy — wyświetlaj dane pozycji jako proste, automatycznie generowane wykresy, twórz własne wykresy lub porównuj wiele pozycji na wykresach tworzonych ad hoc.
- Mapy sieci — twórz wizualne reprezentacje infrastruktury sieciowej z konfigurowalnymi elementami i wskaźnikami.
- Pulpity — dostosowuj układy, łącząc różne widżety pulpitu, które wizualizują dane w różnych formatach.

Wykresy

Przegląd

Przy dużej ilości danych napływających do Zabbix użytkownikom znacznie łatwiej jest obserwować wizualną reprezentację tego, co się dzieje, zamiast analizować wyłącznie liczby.

W tym miejscu przydają się wykresy. Wykresy umożliwiają szybkie zrozumienie przepływu danych, skorelowanie problemów, ustalenie, kiedy coś się rozpoczęło, lub zilustrowanie, kiedy coś może przerodzić się w problem.

Zabbix zapewnia użytkownikom:

- Wbudowane proste wykresy danych jednej pozycji.
- Możliwość tworzenia bardziej złożonych niestandardowych wykresów.
- Szybki dostęp do porównania kilku pozycji za pomocą wykresów ad hoc.
- Nowoczesne, konfigurowalne wektorowe wykresy i wykresy kołowe.

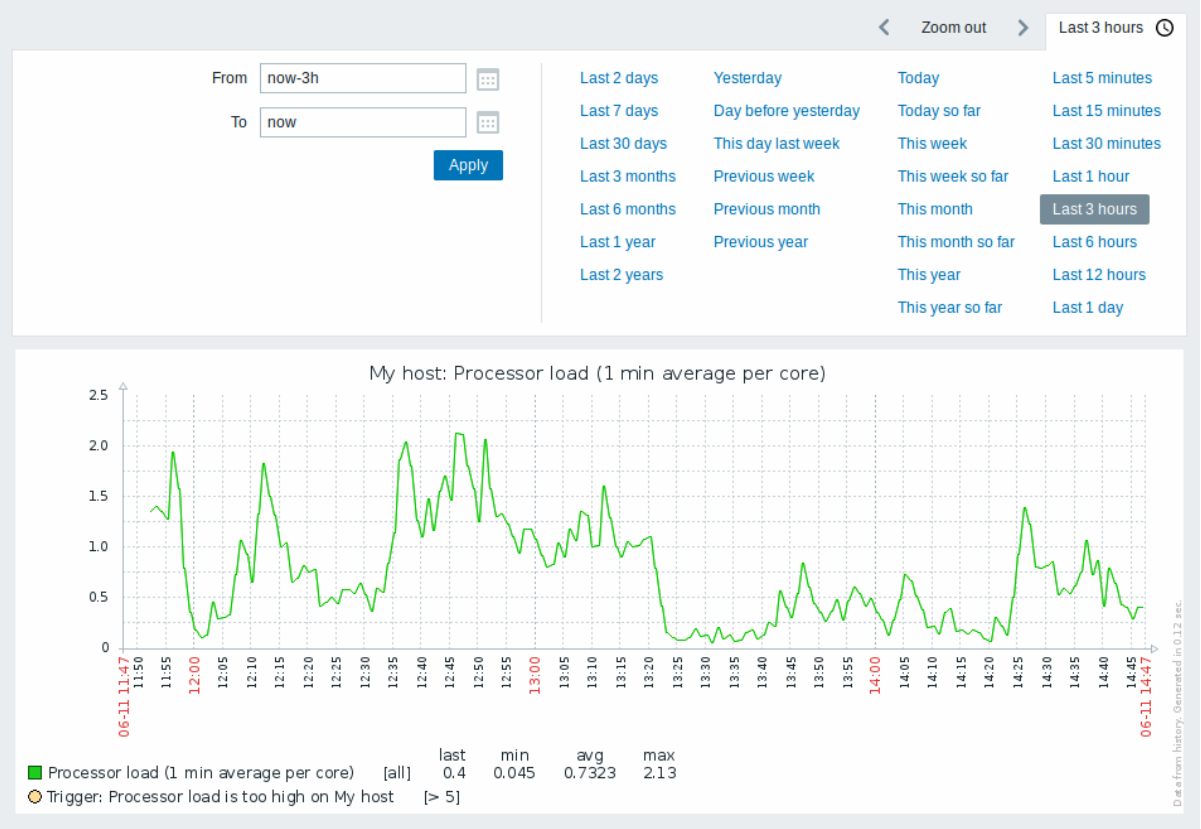
Proste wykresy

Przegląd

Proste wykresy służą do wizualizacji danych zebranych przez pozycje.

Wyświetlanie prostych wykresów nie wymaga od użytkownika żadnej konfiguracji. Są one udostępniane bezpłatnie przez Zabbix.

Wystarczy przejść do sekcji *Monitoring* > *Latest data* i kliknąć link *Graph* przy odpowiedniej pozycji, aby wyświetlić wykres.



Note:
Proste wykresy są dostępne dla wszystkich pozycji numerycznych. W przypadku pozycji tekstowych w sekcji *Monitoring* > *Latest data* dostępny jest link do *History*.

Selektor okresu

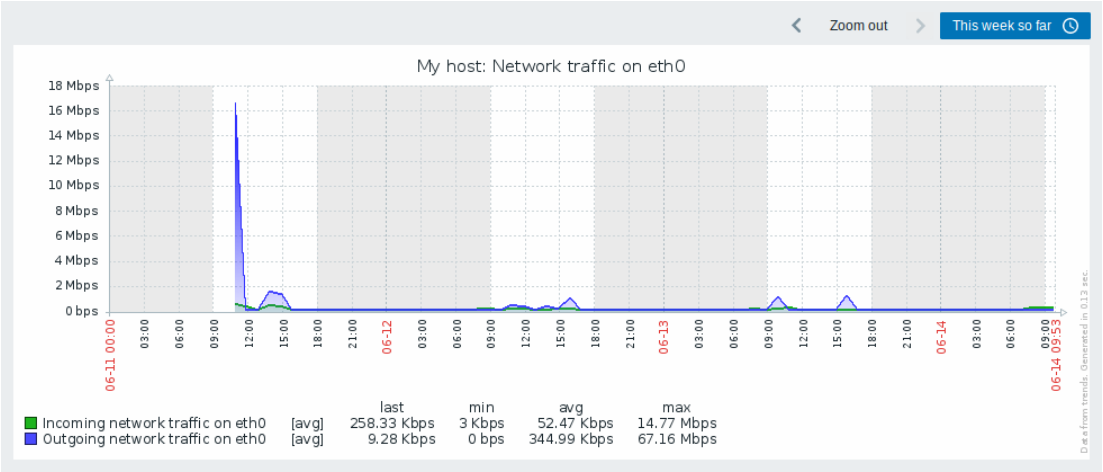
Selektor okresu nad wykresem umożliwia wybranie często używanych okresów jednym kliknięciem myszy. Więcej informacji można znaleźć w sekcji Selektory okresu i hosta.

Ostatnie dane a dłuższe okresy

Dla bardzo ostatnich danych rysowana jest **pojedyncza** linia łącząca każdą otrzymaną wartość. Pojedyncza linia jest rysowana tak długo, jak długo dostępny jest co najmniej jeden poziomy piksel dla jednej wartości.

Dla danych pokazujących dłuższy okres rysowane są **trzy linie** — ciemnozielona pokazuje średnią, natomiast jasnoróżowa i jasnozielona linia pokazują wartości maksymalne i minimalne w danym momencie. Przestrzeń między wartościami wysokimi i niskimi jest wypełniona żółtym tłem.

Czas pracy (dni robocze) jest wyświetlany na wykresach jako białe tło, natomiast czas wolny od pracy jest wyświetlany na szaro (w domyślnym motywie frontend *Original blue*).

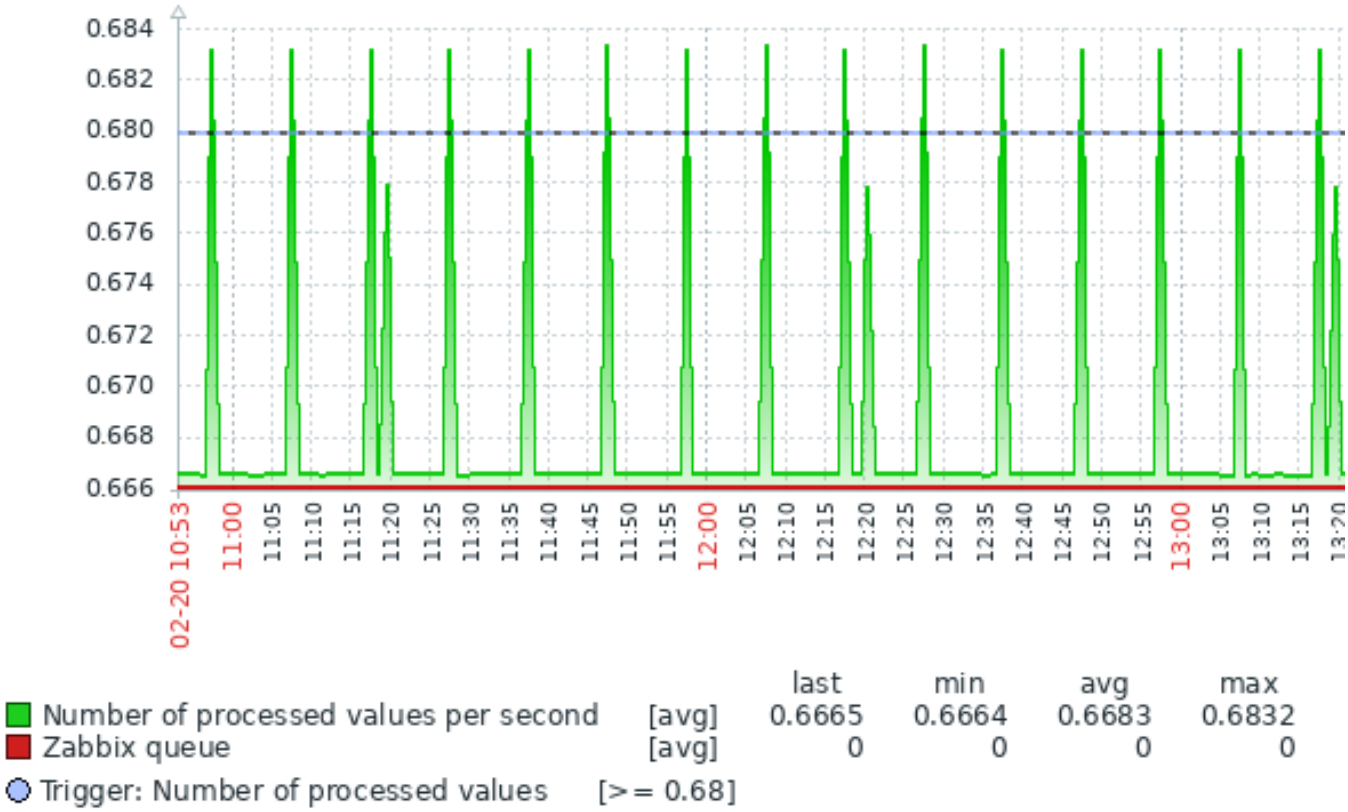


Czas pracy jest zawsze wyświetlany na prostych wykresach, natomiast jego wyświetlanie na wykresach niestandardowych zależy od preferencji użytkownika.

Czas pracy nie jest wyświetlany, jeśli wykres pokazuje okres dłuższy niż 3 miesiące.

Linie wyzwalaczy

Proste wyzwalacze są wyświetlane jako linie z czarnymi kreskami na kolorze ważności wyzwalacza — zwróć uwagę na niebieską linię na wykresie oraz informacje o wyzwalaczu wyświetlane w legendzie. Na wykresie mogą być wyświetlane maksymalnie 3 linie wyzwalaczy; jeśli wyzwalaczy jest więcej, priorytet mają wyzwalacze o niższej ważności. Wyzwalacze są zawsze wyświetlane na prostych wykresach, natomiast ich wyświetlanie na wykresach niestandardowych zależy od preferencji użytkownika.



Generowanie na podstawie historii/trendów

Wykresy mogą być tworzone na podstawie historii lub trendów pozycji.

Użytkownicy, którzy mają aktywowany tryb debugowania frontendu, widzą w prawym dolnym rogu wykresu szary, pionowy opis wskazujący źródło danych.

Na to, czy używana jest historia, czy trendy, wpływa kilka czynników:

- Okres przechowywania historii pozycji. Na przykład historia pozycji może być przechowywana przez 14 dni. W takim przypadku wszystkie dane starsze niż czternaście dni będą pochodzić z trendów.
- Zagęszczenie danych na wykresie. Jeśli liczba sekund przypadających na jeden poziomy piksel wykresu przekracza 3600/16, wyświetlane są dane trendów (nawet jeśli historia pozycji jest nadal dostępna dla tego samego okresu).

- Jeśli trendy są wyłączone, do tworzenia wykresu używana jest historia pozycji — o ile jest dostępna dla danego okresu.

Brak danych

W przypadku pozycji z regularnym interwałem aktualizacji na wykresie nic nie jest wyświetlane, jeśli dane pozycji nie są zbierane.

Jednak w przypadku pozycji trapper oraz pozycji z harmonogramowanym interwałem aktualizacji (i regularnym interwałem aktualizacji ustawionym na 0) rysowana jest prosta linia prowadząca do pierwszej zebranej wartości oraz od ostatniej zebranej wartości do końca wykresu; linia znajduje się odpowiednio na poziomie pierwszej/ostatniej wartości.

Przełączanie na wartości surowe

Lista rozwijana w prawym górnym rogu umożliwia przełączenie z prostego wykresu na listy *Wartości/500 najnowszych wartości*. Może to być przydatne do wyświetlania wartości liczbowych składających się na wykres.

Przedstawione tutaj wartości są surowe, tzn. nie są stosowane jednostki ani przetwarzanie końcowe wartości. Mapowanie wartości jest jednak stosowane.

Znane problemy

Zobacz znane problemy dotyczące wykresów.

Własne wykresy

Przegląd

Niestandardowe wykresy, jak sama nazwa wskazuje, oferują możliwości dostosowywania.

Podczas gdy proste wykresy dobrze nadają się do wyświetlania danych pojedynczej pozycji, nie oferują możliwości konfiguracji.

Dlatego jeśli chcesz zmienić styl wykresu lub sposób wyświetlania linii albo porównać kilka pozycji, na przykład ruch przychodzący i wychodzący na jednym wykresie, potrzebujesz niestandardowego wykresu.

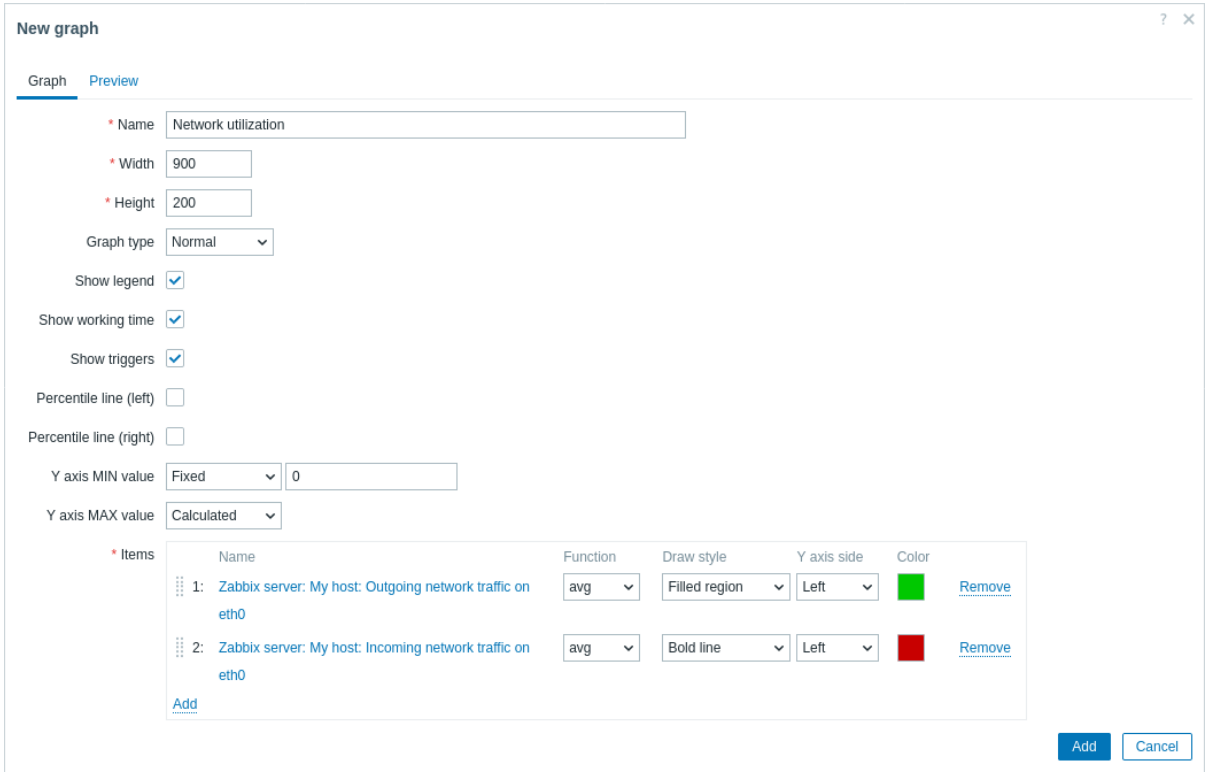
Niestandardowe wykresy są konfigurowane ręcznie.

Można je tworzyć dla jednego hosta lub kilku hostów albo dla pojedynczego szablonu.

Konfigurowanie niestandardowych wykresów

Aby utworzyć niestandardowy wykres, wykonaj następujące czynności:

1. Przejdź do *Zbieranie danych* > *Hosty* (lub *Szablony*).
2. Kliknij *Wykresy* w wierszu obok wybranego hosta lub szablonu.
3. Na ekranie Wykresy kliknij *Utwórz wykres*.
4. Edytuj atrybuty wykresu.



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

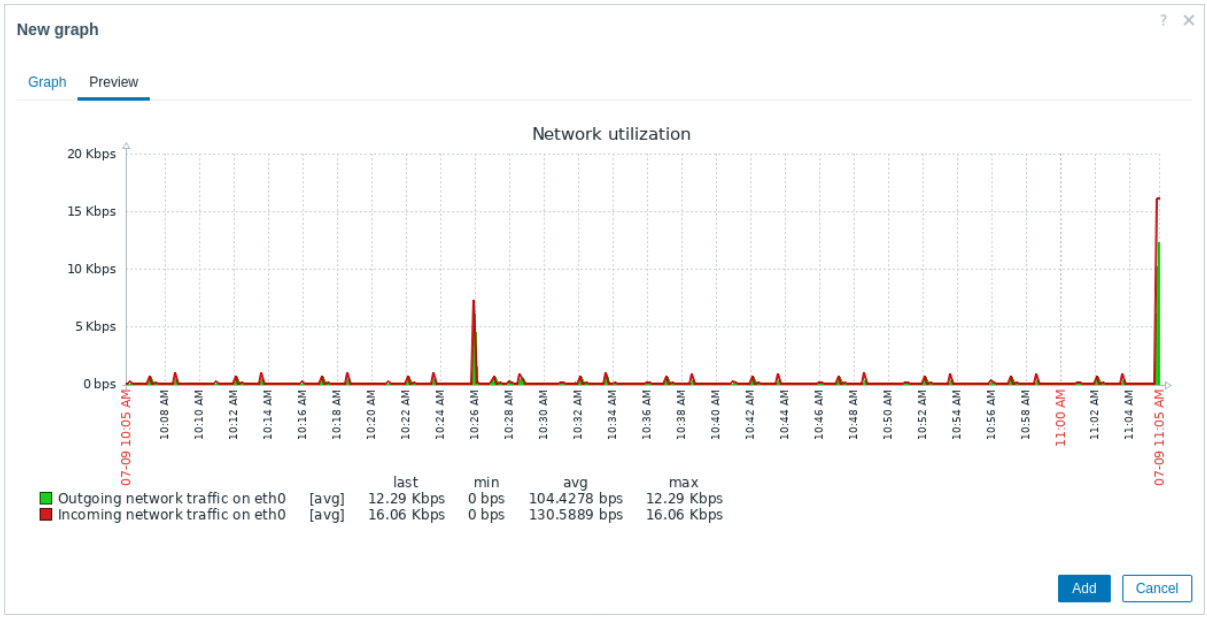
Atrybuty wykresu:

Parameter	Description
Name	Unikalna nazwa wykresu. W tym polu obsługiwane są wyrażenia macros, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`). Makra {HOST.HOST<1-9>} są obsługiwane do użycia w tym makrze, z odwołaniem do pierwszego, drugiego, trzeciego itd. hosta na wykresie, na przykład `{?avg(/{HOST.HOST2}/key,1h)}`. Należy pamiętać, że odwołanie do pierwszego hosta za pomocą tego makra jest zbędne, ponieważ do pierwszego hosta można odwołać się niejawnie, na przykład `{?avg(//key,1h)}`.
Width	Szerokość wykresu w pikselach (tylko dla podglądu oraz wykresów kołowych i eksplodowanych).
Height	Wysokość wykresu w pikselach.
Graph type	Typ wykresu: **Normal** - zwykły wykres, wartości wyświetlane jako linie **Stacked** - wykres skumulowany, wyświetlane wypełnione obszary **Pie** - wykres kołowy **Exploded** - wykres kołowy "eksplodowany", fragmenty wyświetlane jako "wycięte" z koła
Show legend	Zaznaczenie tego pola spowoduje wyświetlenie legendy wykresu.
Show working time	Jeśli zaznaczone, godziny niepracujące będą wyświetlane na szarym tle. Ten parametr nie jest dostępny dla wykresów kołowych i eksplodowanych.
Show triggers	Jeśli zaznaczone, proste wyzwalacze będą wyświetlane jako linie z czarnymi kreskami na kolorze odpowiadającym poziomowi problemu wyzwalacza. Ten parametr nie jest dostępny dla wykresów kołowych i eksplodowanych.
Percentile line (left)	Wyświetla percentyl dla lewej osi Y. Jeśli na przykład ustawiono percentyl 95%, linia percentyla będzie na poziomie, poniżej którego znajduje się 95 procent wartości. Wyświetlana jako jasnozielona linia. Dostępne tylko dla zwykłych wykresów.
Percentile line (right)	Wyświetla percentyl dla prawej osi Y. Jeśli na przykład ustawiono percentyl 95%, linia percentyla będzie na poziomie, poniżej którego znajduje się 95 procent wartości. Wyświetlana jako jaskrawoczerwona linia. Dostępne tylko dla zwykłych wykresów.
Y axis MIN value	Minimalna wartość osi Y: **Calculated** - minimalna wartość osi Y zostanie obliczona automatycznie. **Fixed** - stała minimalna wartość osi Y. **Item** - ostatnia wartość wybranej pozycji będzie wartością minimalną. Ten parametr nie jest dostępny dla wykresów kołowych i eksplodowanych.
Y axis MAX value	Maksymalna wartość osi Y: **Calculated** - maksymalna wartość osi Y zostanie obliczona automatycznie. **Fixed** - stała maksymalna wartość osi Y. **Item** - ostatnia wartość wybranej pozycji będzie wartością maksymalną Ten parametr nie jest dostępny dla wykresów kołowych i eksplodowanych.
3D view	Włącza styl 3D. Tylko dla wykresów kołowych i eksplodowanych.
Items	Pozycje, których dane mają być wyświetlane na tym wykresie. Kliknij *Dodaj*, aby wybrać pozycje. Możesz także wybrać różne opcje wyświetlania (funkcja, styl rysowania, wyświetlanie lewej/prawej osi, kolor).
Sort order (0→100)	Kolejność rysowania. 0 będzie przetwarzane jako pierwsze. Można użyć do rysowania linii lub obszarów za innym elementem (lub przed nim). Możesz przeciągać i upuszczać pozycje za pomocą ikony na początku wiersza, aby ustawić kolejność sortowania lub określić, która pozycja jest wyświetlana przed inną.
Name	Nazwa wybranej pozycji jest wyświetlana jako link. Kliknięcie linku otwiera listę innych dostępnych pozycji.

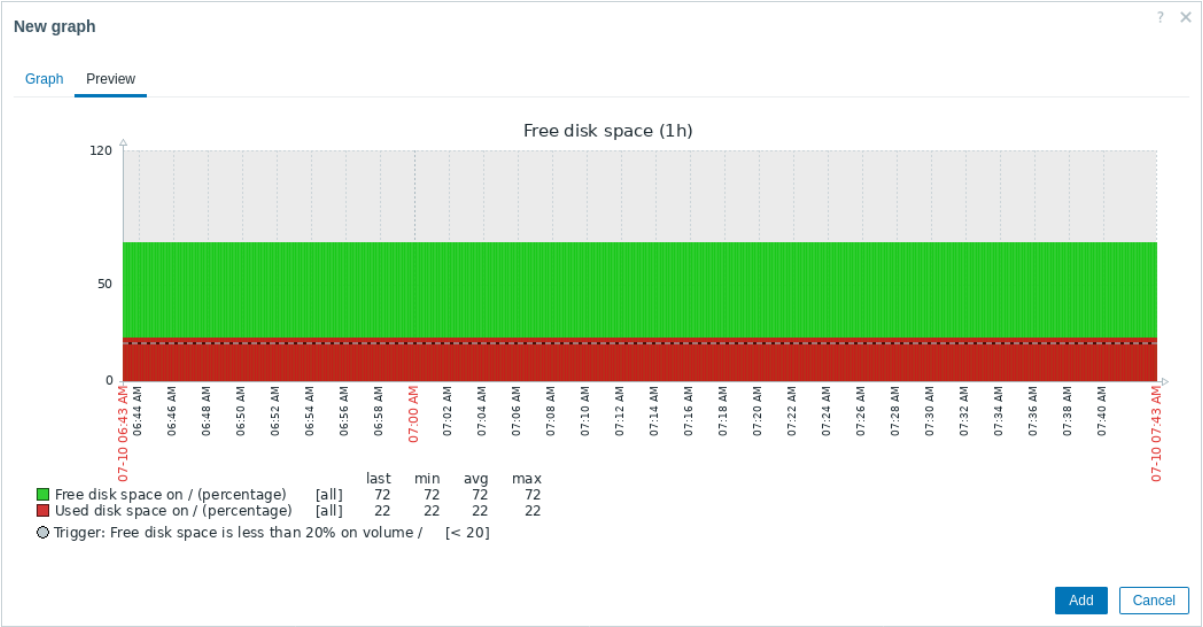
Parameter	Description
Type	Typ (dostępny tylko dla wykresów kołowych i eksplodowanych): **Simple** - wartość pozycji jest przedstawiana proporcjonalnie na wykresie kołowym **Graph sum** - wartość pozycji reprezentuje cały wykres kołowy Należy pamiętać, że kolorowanie pozycji ”graph sum” będzie widoczne tylko w takim zakresie, w jakim nie jest ono zajęte przez pozycje ”proportional”.
Function	Wybierz, które wartości będą wyświetlane, gdy dla jednej pozycji istnieje więcej niż jedna wartość na pojedynczy pionowy piksel wykresu: **all** - wyświetla wszystkie możliwe wartości (minimum, maksimum, średnia) na wykresie. Należy pamiętać, że dla krótszych okresów to ustawienie nie ma wpływu; dopiero dla dłuższych okresów, gdy zagęszczenie danych w pionowym pikselu wykresu wzrasta, opcja ’all’ zaczyna wyświetlać wartości minimalne, maksymalne i średnie. Ta funkcja jest dostępna tylko dla typu wykresu *Normal*. Zobacz też: Generowanie wykresów z historii/trendów. **avg** - wyświetla wartości średnie **last** - wyświetla najnowsze wartości. Ta funkcja jest dostępna tylko wtedy, gdy jako typ wykresu wybrano *Pie/Exploded pie*. **max** - wyświetla wartości maksymalne **min** - wyświetla wartości minimalne
Draw style	Wybierz styl rysowania (dostępny tylko dla zwykłych wykresów; dla wykresów skumulowanych zawsze używany jest wypełniony obszar), który ma zostać zastosowany do danych pozycji - *Line*, *Bold line*, *Filled region*, *Dot*, *Dashed line*, *Gradient line*.
Y axis side	Wybierz stronę osi Y, na której mają być wyświetlane dane pozycji - *Left*, *Right*.
Color	Wybierz kolor, który ma zostać zastosowany do danych pozycji.

Podgląd wykresu

Na karcie *Podgląd* wyświetlany jest podgląd wykresu, dzięki czemu możesz od razu zobaczyć, co tworzysz.



Pamiętaj, że podgląd nie będzie wyświetlał żadnych danych dla pozycji szablonu.



W tym przykładzie zwróć uwagę na pogrubioną linię przerywaną przedstawiającą poziom wyzwalacza oraz informacje o wyzwalaczu wyświetlane w legendzie.

Note:
Można wyświetlić nie więcej niż 3 linie wyzwalaczy. Jeśli wyzwalaczy jest więcej, do wyświetlenia priorytetowo wybierane są wyzwalacze o niższym poziomie ważności.

 Jeśli wysokość wykresu jest ustawiona na mniej niż 120 pikseli, w legendzie nie będzie wyświetlany żaden wyzwalacz.

Wykresy ad hoc

Przegląd

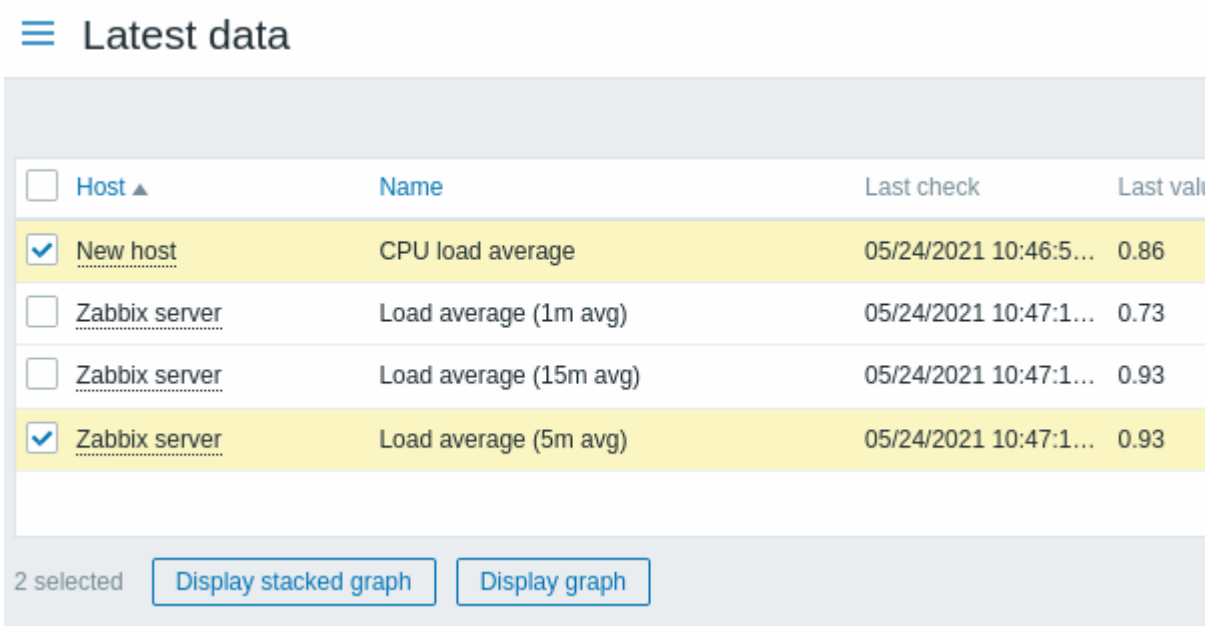
Chociaż prosty wykres jest przydatny do wyświetlania danych z pojedynczej pozycji, a wykresy niestandardowe oferują opcje dostosowywania, żadne z tych rozwiązań nie umożliwia szybkiego utworzenia wykresu porównawczego dla wielu pozycji przy minimalnym nakładzie pracy i konserwacji.

Aby rozwiązać ten problem, można w bardzo szybki sposób tworzyć wykresy ad hoc dla kilku pozycji.

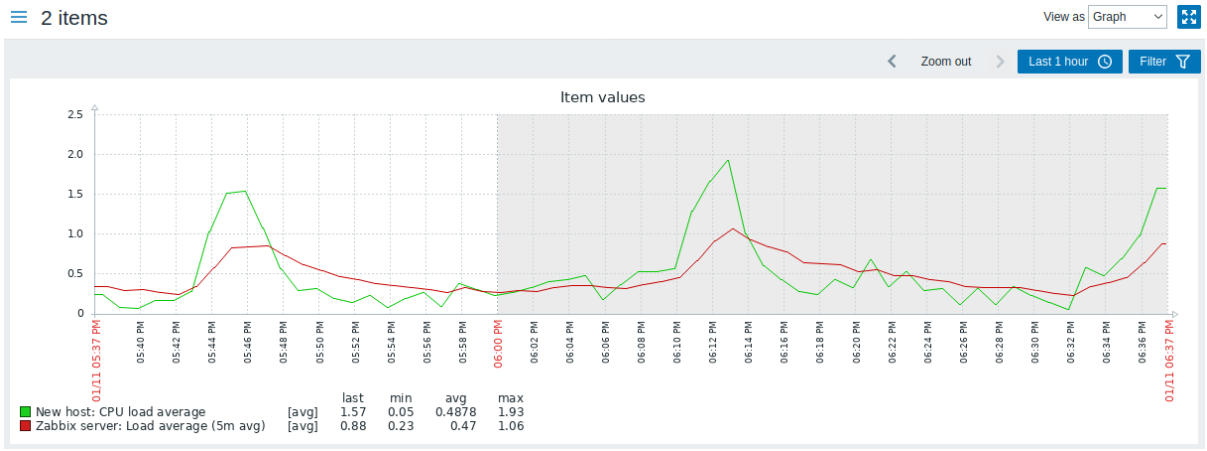
Konfiguracja

Aby utworzyć wykres ad hoc, wykonaj następujące czynności:

1. Przejdź do *Monitoring* > *Latest data*.
2. Użyj filtra, aby wyświetlić interesujące Cię pozycje.
3. Zaznacz pola wyboru przy pozycjach, które chcesz przedstawić na wykresie.
4. Kliknij przyciski *Display stacked graph* lub *Display graph*.

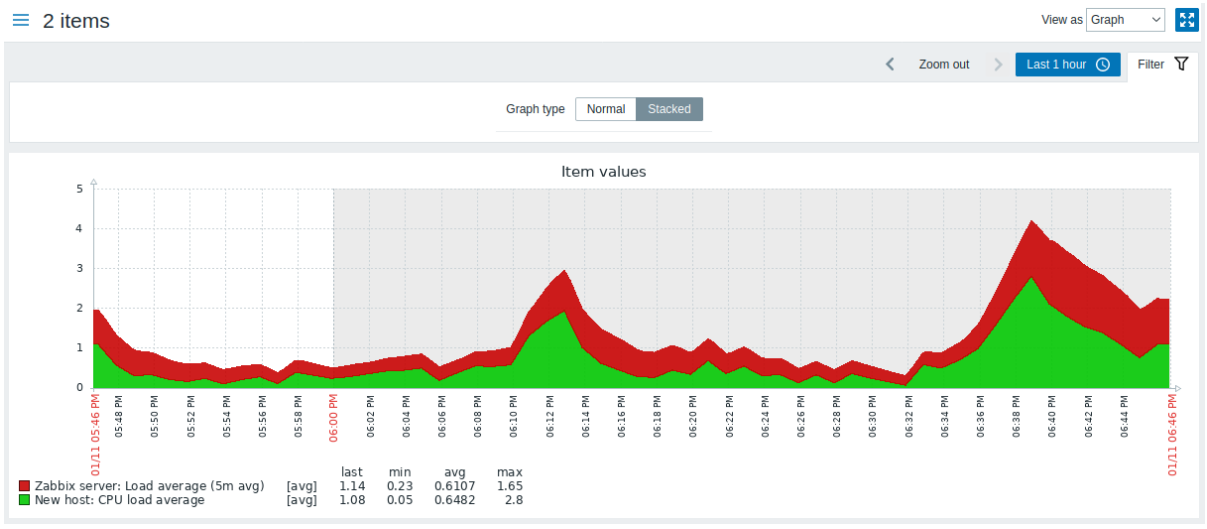


Wykres zostanie utworzony natychmiast:



Należy pamiętać, że aby uniknąć wyświetlania zbyt wielu linii na wykresie, dla każdej pozycji wyświetlana jest tylko wartość średnia (linie wartości min./maks. nie są wyświetlane). Wyzwalacze oraz informacje o wyzwalaczach nie są wyświetlane na wykresie.

W oknie utworzonego wykresu dostępny jest selektor okresu czasu oraz możliwość przełączenia z wykresu liniowego „normalnego” na wykres skumulowany i z powrotem.



Mapy sieciowe

Przegląd

Jeśli zarządzasz siecią, możesz chcieć mieć gdzieś ogólny widok swojej infrastruktury. W tym celu możesz tworzyć mapy w Zabbix — sieci oraz czegokolwiek zechcesz.

Wszyscy użytkownicy mogą tworzyć mapy sieci. Mapy mogą być publiczne (dostępne dla wszystkich użytkowników) lub prywatne (dostępne dla wybranych użytkowników).

Przejdź do konfigurowania mapy sieci.

Konfigurowanie mapy sieciowej

Omówienie

Konfigurowanie mapy w Zabbix wymaga najpierw utworzenia mapy przez zdefiniowanie jej ogólnych parametrów, a następnie rozpoczęcia wypełniania samej mapy elementami i ich połączeniami.

Mapę można wypełnić elementami takimi jak host, grupa hostów, wyzwalacz, obraz lub inna mapa.

Ikony służą do reprezentowania elementów mapy. Możesz zdefiniować informacje, które będą wyświetlane wraz z ikonami, oraz ustawić, aby ostatnie problemy były wyświetlane w specjalny sposób. Możesz łączyć ikony i definiować informacje wyświetlane na połączeniach.

Możesz dodać niestandardowe adresy URL, które będą dostępne po kliknięciu ikon. W ten sposób możesz powiązać ikonę hosta z właściwościami hosta lub ikonę mapy z inną mapą.

Liczba problemów na mapach jest wyświetlana tylko dla problemów przyczynowych.

Mapy są zarządzane w *Monitoring* > *Maps*, gdzie można je konfigurować, zarządzać nimi i przeglądać je. W widoku monitorowania możesz klikać ikony i korzystać z odnośników do niektórych skryptów i adresów URL.

Mapy sieciowe są oparte na grafice wektorowej (SVG).

Mapy publiczne i prywatne

Wszyscy użytkownicy Zabbix (w tym użytkownicy niebędący administratorami) mogą tworzyć mapy sieci. Mapy mają właściciela — użytkownika, który je utworzył. Mapy mogą być publiczne lub prywatne.

- Mapy *publiczne* są widoczne dla wszystkich użytkowników, jednak aby je zobaczyć, użytkownik musi mieć uprawnienia do odczytu co najmniej jednego elementu mapy. Mapy publiczne mogą być edytowane, jeśli użytkownik/grupa użytkowników ma uprawnienia do odczytu i zapisu do tej mapy oraz co najmniej uprawnienia do odczytu do wszystkich elementów odpowiedniej mapy, w tym wyzwalaczy w połączeniach.
- Mapy *prywatne* są widoczne tylko dla ich właściciela oraz użytkowników/grup użytkowników, którym mapa została udostępniona przez właściciela. Zwykli użytkownicy (niebędący Super adminami) mogą udostępniać mapy tylko grupom, do których należą, oraz użytkownikom należącym do tych grup. Użytkownicy na poziomie Admin mogą widzieć mapy prywatne niezależnie od tego, czy są ich właścicielami, czy należą do listy użytkowników, którym mapa została udostępniona. Mapy prywatne mogą być edytowane przez właściciela mapy oraz wtedy, gdy użytkownik/grupa użytkowników ma uprawnienia do odczytu i zapisu do tej mapy oraz co najmniej uprawnienia do odczytu do wszystkich elementów odpowiedniej mapy, w tym wyzwalaczy w połączeniach.

Elementy mapy, do których użytkownik nie ma uprawnień odczytu, są wyświetlane z wyszarzoną ikoną, a wszystkie informacje tekstowe o elemencie są ukryte. Jednak etykieta wyzwalacza jest widoczna, nawet jeśli użytkownik nie ma uprawnień do wyzwalacza.

Aby dodać element do mapy, użytkownik musi również mieć co najmniej uprawnienia do odczytu tego elementu.

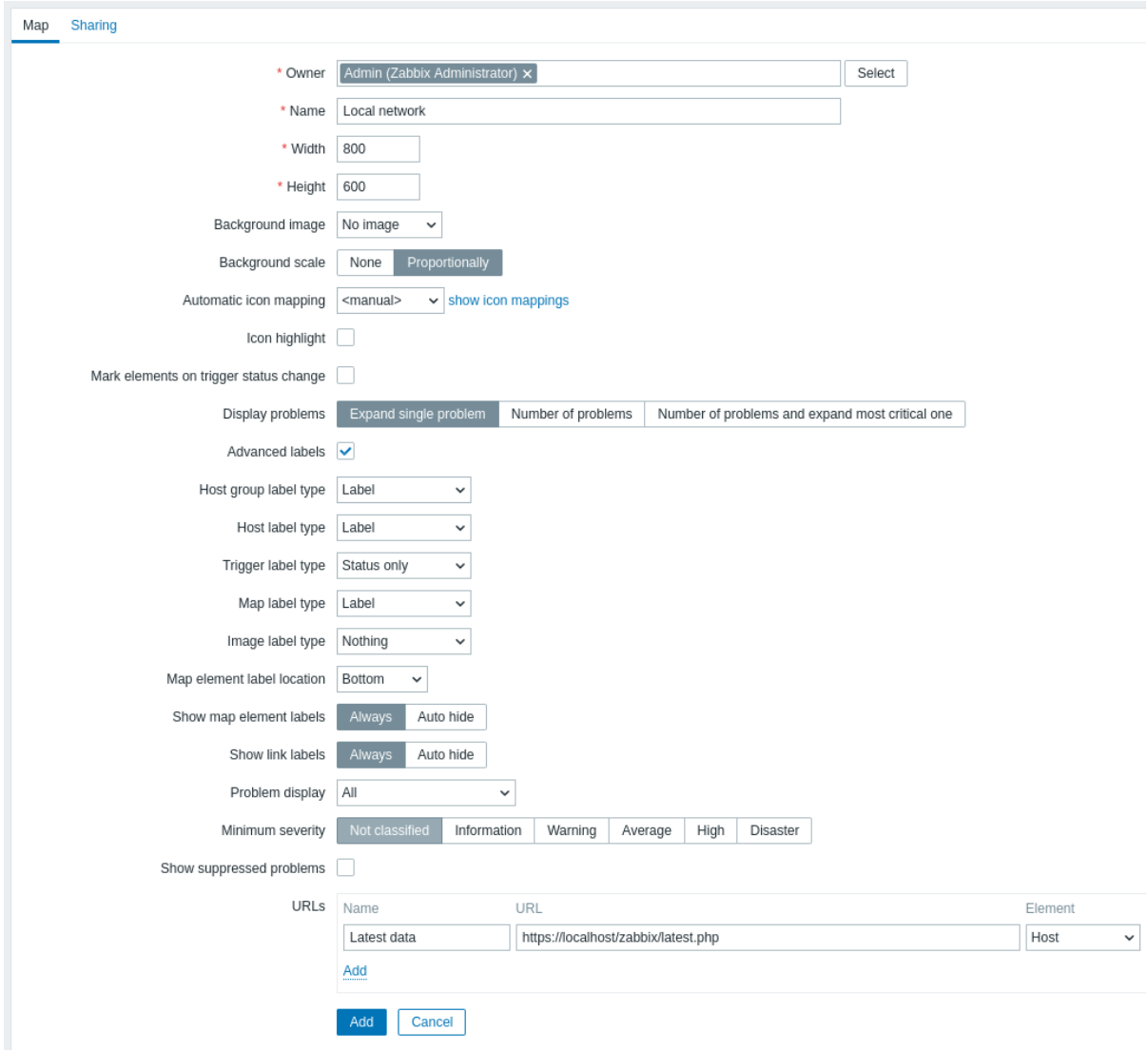
Tworzenie mapy

Aby utworzyć mapę:

1. Przejdź do *Monitoring* > *Maps*.
2. Przejdź do widoku zawierającego wszystkie mapy.
3. Kliknij *Create map*.
4. Skonfiguruj szczegóły mapy.

Możesz również użyć przycisku *Clone* w formularzu konfiguracji istniejącej mapy, aby utworzyć nową mapę. Nowa mapa będzie mieć wszystkie właściwości istniejącej mapy, w tym ogólne atrybuty układu, a także elementy istniejącej mapy.

Karta **Map** zawiera ogólne atrybuty mapy:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

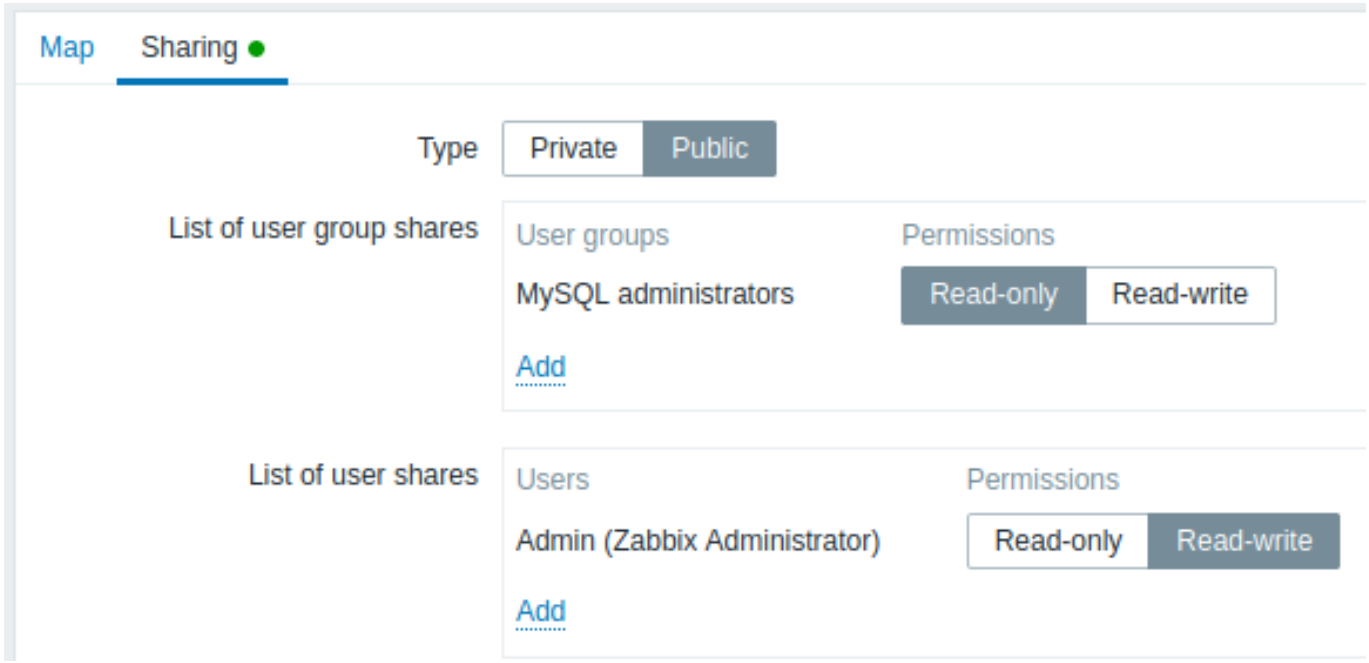
Ogólne atrybuty mapy:

Parameter	Description
Owner	Nazwa właściciela mapy.
Name	Unikalna nazwa mapy.
Width	Szerokość mapy w pikselach.
Height	Wysokość mapy w pikselach.
Background image	Użyj obrazu tła: **No image** - brak obrazu tła (białe tło) **Image** - wybrany obraz, który zostanie użyty jako obraz tła. Możesz użyć mapy geograficznej lub dowolnego innego obrazu, aby wzbogacić mapę.
Background scale	Użyj skalowania tła: **None** - brak skalowania; **Proportionally** - wypełnienie całego tła mapy przy zachowaniu proporcji obrazu (domyślnie).
Automatic icon mapping	Możesz włączyć automatyczne mapowanie ikon skonfigurowane w *Administration* > *General* > *Icon mapping*. Mapowanie ikon umożliwia przypisywanie określonych ikon do określonych pól inwentaryzacji hosta.
Icon highlight	Jeśli zaznaczysz to pole wyboru, elementy mapy będą wyróżniane. Elementy z aktywnym wyzwalaczem otrzymają okrągłe tło w kolorze wyzwalacza o najwyższym poziomie ważności. Ponadto wokół okręgu zostanie wyświetlona gruba zielona linia, jeśli wszystkie problemy zostały potwierdzone. Elementy ze stanem „disabled” lub „in maintenance” otrzymają odpowiednio kwadratowe tło w kolorze szarym lub pomarańczowym. Zobacz także: Wyświetlanie map

Parameter	Description
Mark elements on trigger status change	Niedawna zmiana stanu wyzwalacza (nowy problem lub rozwiązanie problemu) zostanie wyróżniona znacznikami (czerwonymi trójkątami skierowanymi do wewnątrz) na trzech bokach ikony elementu, które nie są zajęte przez etykietę. Znaczniki są wyświetlane przez 30 minut.
Display problems	Wybierz sposób wyświetlania problemów przy elemencie mapy: **Expand single problem** - jeśli istnieje tylko jeden problem, wyświetlana jest jego nazwa. W przeciwnym razie wyświetlana jest łączna liczba problemów. **Number of problems** - wyświetlana jest łączna liczba problemów **Number of problems and expand most critical one** - wyświetlana jest nazwa najpoważniejszego problemu oraz łączna liczba problemów. „Najpoważniejszy" problem jest określany na podstawie poziomu ważności problemu, a w przypadku równego poziomu - na podstawie identyfikatora zdarzenia problemu (problem o wyższym identyfikatorze lub późniejszy jest wyświetlany jako pierwszy). W przypadku *elementu mapy wyzwalacza* jest to określane na podstawie poziomu ważności problemu, a w przypadku równego poziomu - na podstawie pozycji wyzwalacza na liście wyzwalaczy. Jeśli występuje wiele problemów dotyczących tego samego wyzwalacza, wyświetlany jest najnowszy z nich.
Advanced labels	Jeśli zaznaczysz to pole wyboru, będzie można definiować osobne typy etykiet dla poszczególnych typów elementów.
Host group label type	Typ etykiety używany dla elementu mapy: **Label** - etykieta elementu mapy **IP address** - adres IP **Element name** - nazwa elementu (na przykład nazwa hosta) **Status only** - tylko stan (OK lub PROBLEM) **Nothing** - etykiety nie są wyświetlane
Host label type *Trigger label type* *Map label type* *Image label type*	
Map element label location	Położenie etykiety względem elementu mapy: **Bottom** - pod elementem mapy **Left** - po lewej stronie **Right** - po prawej stronie **Top** - nad elementem mapy
Show map element labels	Wybierz sposób wyświetlania etykiet elementów mapy: **Always** - zawsze wyświetlane (domyślnie); **Auto hide** - ukrywanie etykiety, gdy kursor nie znajduje się nad nią i nie jest ona zaznaczona.
Show link labels	Wybierz sposób wyświetlania etykiet połączeń: **Always** - zawsze wyświetlane (domyślnie); **Auto hide** - ukrywanie etykiety, gdy kursor nie znajduje się nad nią i nie jest ona zaznaczona.
Problem display	Wyświetlaj liczbę problemów jako: **All** - wyświetlana jest pełna liczba problemów **Separated** - liczba niepotwierdzonych problemów jest wyświetlana osobno jako część całkowitej liczby problemów **Unacknowledged only** - wyświetlana jest tylko liczba niepotwierdzonych problemów
Minimum severity	Problemy poniżej wybranego minimalnego poziomu ważności nie będą wyświetlane na mapie. Na przykład po wybraniu poziomu *Warning* zmiany dotyczące wyzwalaczy o poziomie *Information* i *Not classified* nie będą odzwierciedlane na mapie.
Show suppressed problems	Zaznacz to pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby ukryte (niewyświetlane) z powodu konserwacji hosta.
URLs	W tym miejscu można zdefiniować adresy URL dla każdego typu elementu, gdzie: **Name** - etykieta adresu URL; **URL** - adres URL (do 2048 znaków); **Element** - typ elementu (np. *Host*, *Host group* itd.). Będą one wyświetlane jako łącza po kliknięciu elementu przez użytkownika w trybie wyświetlania mapy. W nazwach i wartościach adresów URL mapy można używać makr. Pełną listę można znaleźć w sekcji obsługiwane makra, wyszukując hasło „map URL names and values".

Udostępnianie

Karta **Udostępnianie** zawiera typ mapy oraz opcje udostępniania (grupy użytkowników, użytkownicy) dla map prywatnych:



Parameter	Description
Type	Wybierz typ mapy: **Private** - mapa jest widoczna tylko dla wybranych grup użytkowników i użytkowników **Public** - mapa jest widoczna dla wszystkich
List of user group shares	Wybierz grupy użytkowników, dla których mapa jest dostępna. Możesz zezwolić na dostęp tylko do odczytu lub do odczytu i zapisu.
List of user shares	Wybierz użytkowników, dla których mapa jest dostępna. Możesz zezwolić na dostęp tylko do odczytu lub do odczytu i zapisu.

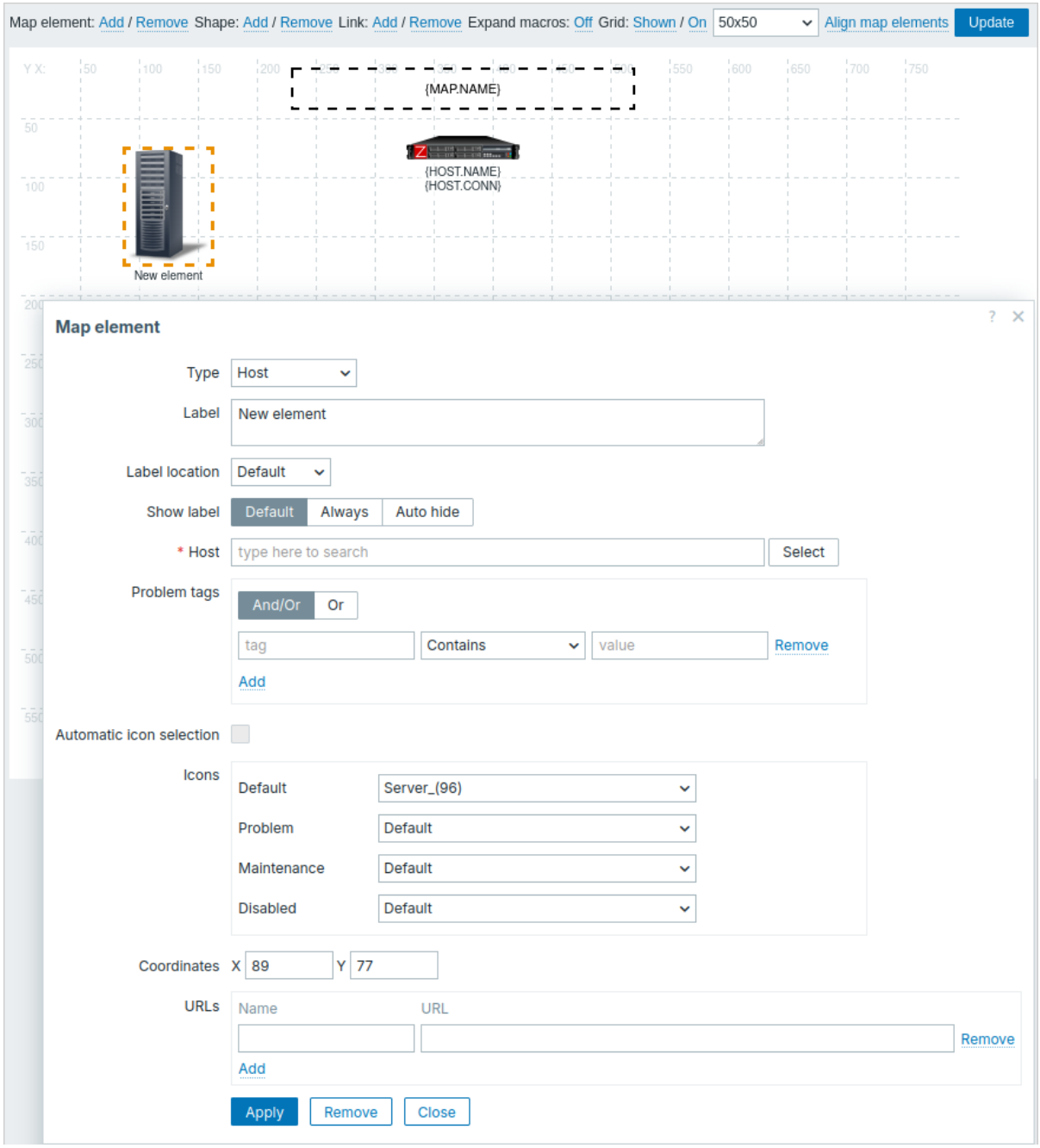
Po kliknięciu *Add*, aby zapisać tę mapę, zostanie utworzona pusta mapa z nazwą, wymiarami i określonymi preferencjami. Teraz trzeba dodać kilka elementów. W tym celu kliknij *Edit* na liście map, aby otworzyć obszar edycji.

Dodawanie elementów

Aby dodać element, kliknij *Add* obok `Map element`. Nowy element pojawi się w lewym górnym rogu mapy. Przeciągnij go i upuść w dowolnym miejscu.

Zwróć uwagę, że przy opcji Grid ustawionej na ”On” elementy będą zawsze wyrównywane do siatki (możesz wybrać różne rozmiary siatki z listy rozwijanej, a także ukryć/pokazać siatkę). Jeśli chcesz umieszczać elementy w dowolnym miejscu bez wyrównywania, ustaw opcję na ”Off”. (Później możesz wyrównać losowe elementy do siatki, klikając *Align map elements*.)

Teraz, gdy masz już kilka elementów na mapie, możesz zacząć je rozróżniać, nadając im nazwy itp. Po kliknięciu elementu wyświetlany jest formularz, w którym możesz ustawić typ elementu, nadać nazwę, wybrać inną ikonę itp.



Atrybuty elementu mapy:

Parameter	Description
Type	Typ elementu: **Host** - ikona reprezentująca stan wszystkich wyzwalaczy wybranego hosta **Map** - ikona reprezentująca stan wszystkich elementów mapy **Trigger** - ikona reprezentująca stan jednego lub wielu wyzwalaczy **Host group** - ikona reprezentująca stan wszystkich wyzwalaczy wszystkich hostów należących do wybranej grupy **Image** - ikona, niepowiązana z żadnym zasobem
Label	Etykieta ikony, dowolny ciąg znaków. Można używać makr i ciągów wielowierszowych. Makra wyrażeń macros są obsługiwane w tym polu, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`). Pełną listę obsługiwanych makr znajdziesz w sekcji supported macros, wyszukując hasło 'map element labels'.

Parameter	Description
Label location	Położenie etykiety względem ikony: **Default** - domyślne położenie etykiety mapy; **Bottom** - pod ikoną; **Left** - po lewej; **Right** - po prawej; **Top** - nad ikoną.
Show label	Wybierz sposób wyświetlania etykiety elementu: **Default** - użyj ustawień *Show map element labels* z konfiguracji mapy (domyślnie); **Always** - zawsze wyświetlana (domyślnie); **Auto hide** - ukryj etykietę, jeśli nie jest wskazywana kursorem ani zaznaczona.
Host	Wybierz host; alternatywnie wpisz nazwę hosta. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona lista rozwijana pasujących hostów. Przewiń w dół, aby wybrać jeden z nich. Kliknij ikonę usuwania (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Host*.
Map	Wybierz mapę; alternatywnie wpisz nazwę mapy. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy mapy zostanie wyświetlona lista rozwijana pasujących map. Przewiń w dół, aby wybrać jedną z nich. Kliknij ikonę usuwania (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Map*.
Triggers	Wybierz jeden lub więcej wyzwalaczy w polu *New triggers* poniżej i kliknij *Add*.Alternatywnie zacznij wpisywać nazwę wyzwalacza i wybierz jeden z listy rozwijanej pasujących wyzwalaczy. Kliknij ikonę usuwania (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Trigger*. Kolejność wybranych wyzwalaczy można zmieniać, ale tylko w obrębie tej samej istotności wyzwalaczy. Wybór wielu wyzwalaczy wpływa również na rozwiązywanie makr {HOST.*} zarówno w trybie edycji, jak i podglądu. W trybie edycji pierwsze wyświetlane makra {HOST.*} będą rozwiązywane w zależności od pierwszego wyzwalacza na liście (na podstawie istotności wyzwalacza). Tryb podglądu zależy od parametru Display problems w ogólnych atrybutach mapy: - Jeśli wybrano tryb *Expand single problem*, pierwsze wyświetlane makra {HOST.*} będą rozwiązywane w zależności od ostatnio wykrytego wyzwalacza problemu (niezależnie od istotności) lub pierwszego wyzwalacza na liście (jeśli nie wykryto problemu). - Jeśli wybrano tryb *Number of problems and expand most critical one*, pierwsze wyświetlane makra {HOST.*} będą rozwiązywane w zależności od istotności wyzwalacza.
Host group	Wybierz grupę hostów; alternatywnie wpisz grupę hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy hostów zostanie wyświetlona lista rozwijana pasujących grup hostów. Przewiń w dół, aby wybrać jedną z nich. Kliknij ikonę usuwania (x), aby usunąć wybór. To pole jest dostępne tylko dla elementów typu *Host group*.
Problem tags	Określ tagi, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek To pole jest dostępne tylko dla elementów typu *Host* i *Host group*.
Automatic icon selection	Gdy pole wyboru jest zaznaczone, mapowanie ikon będzie używane do określenia, która ikona ma być wyświetlana.
Icons	Możesz wybrać wyświetlanie różnych ikon dla elementu w następujących przypadkach: *Default*, *Problem*, *Maintenance* i *Disabled*. To pole jest dostępne tylko dla elementów typu *Host*.

Parameter	Description
Coordinates	Współrzędne **X** i **Y** elementu mapy.
URLs	Tutaj można ustawić dla elementu adresy URL specyficzne dla elementu, gdzie: **Name** - etykieta adresu URL; **URL** - adres URL (do 2048 znaków). Będą one wyświetlane jako linki, gdy użytkownik kliknie element w trybie podglądu mapy. Jeśli element ma własne adresy URL, a dla jego typu zdefiniowano adresy URL na poziomie mapy, zostaną one połączone w tym samym menu. W nazwach i wartościach elementów mapy można używać makr. Pełną listę znajdziesz w sekcji supported macros, wyszukując hasło 'map URL names and values'.

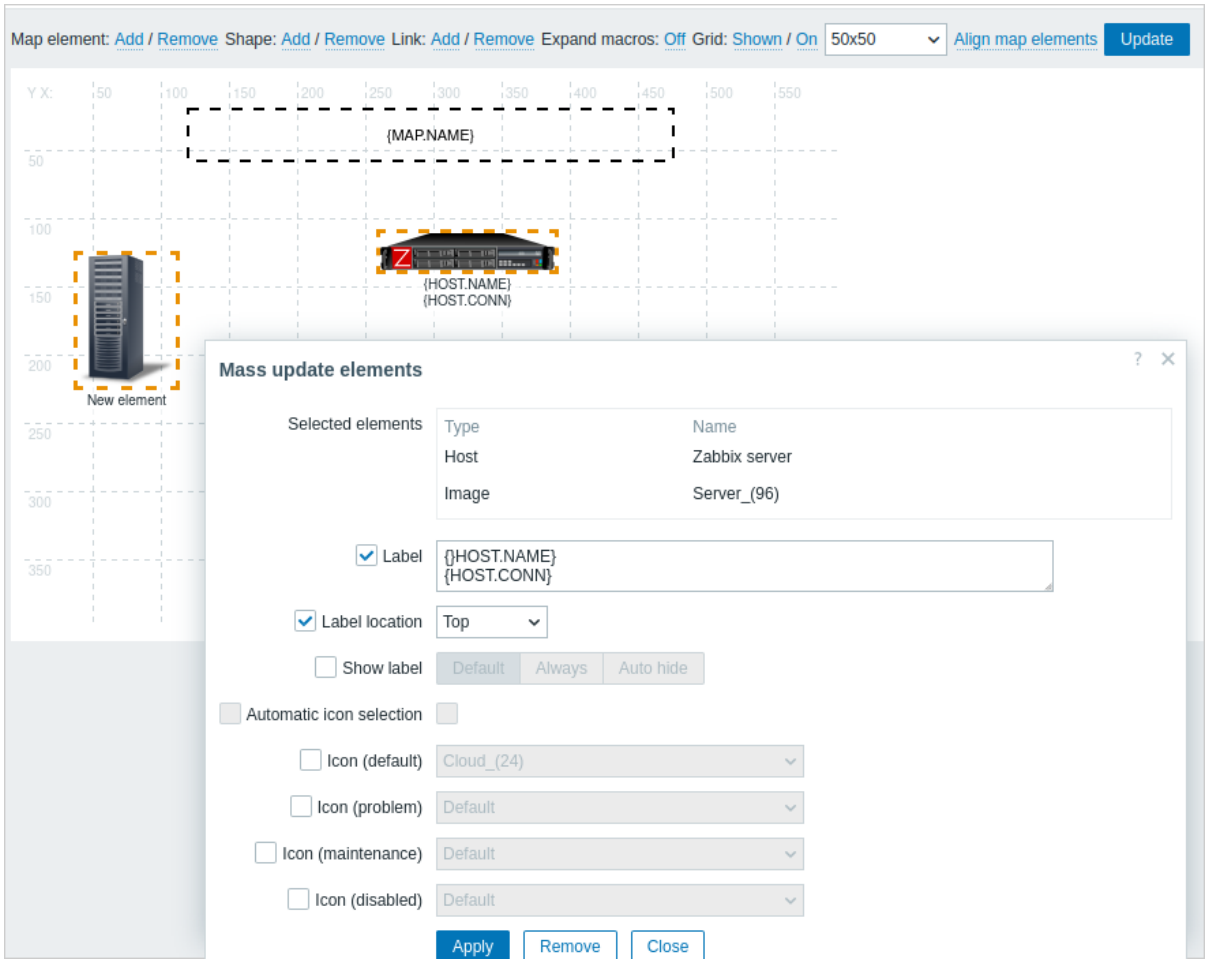
Attention:
Dodane elementy nie są zapisywane automatycznie. Jeśli opuścisz stronę, wszystkie zmiany mogą zostać utracone. Dlatego warto kliknąć przycisk **Update** w prawym górnym rogu. Po kliknięciu zmiany zostaną zapisane niezależnie od tego, co wybierzesz w następnym oknie podręcznym.
Wybrane opcje siatki są również zapisywane wraz z każdą mapą.

Wybieranie elementów

Aby wybrać elementy, wybierz jeden z nich, a następnie przytrzymaj *Ctrl*, aby wybrać pozostałe.

Możesz również wybrać wiele elementów, przeciągając prostokąt w edytowalnym obszarze i zaznaczając wszystkie elementy, które się w nim znajdują.

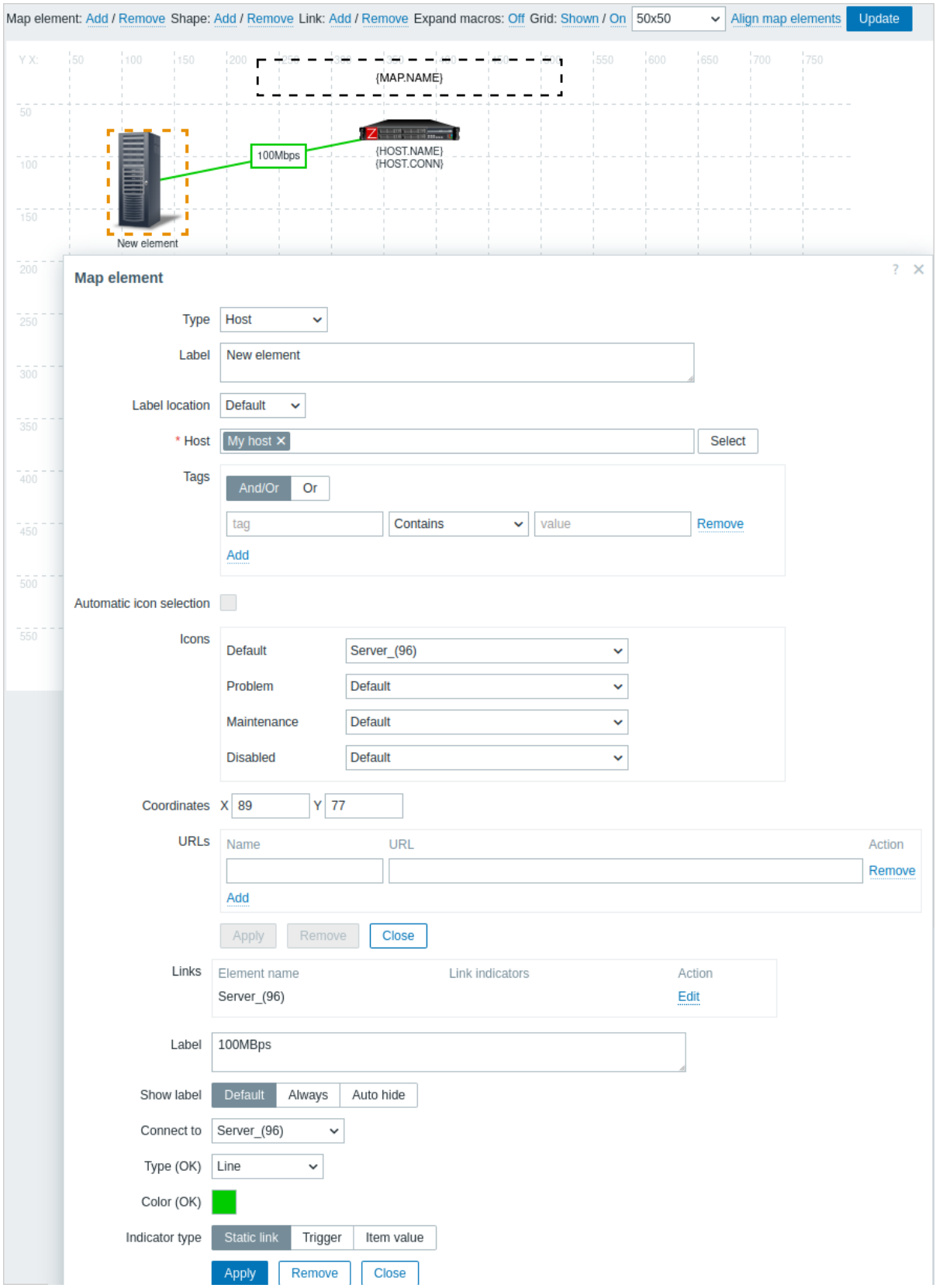
Po wybraniu więcej niż jednego elementu formularz właściwości elementu przełącza się w tryb masowej aktualizacji, dzięki czemu możesz jednocześnie zmienić atrybuty wybranych elementów. Aby to zrobić, zaznacz atrybut za pomocą pola wyboru i wprowadź dla niego nową wartość. Możesz tutaj używać makr (na przykład {HOST.NAME} jako etykiety elementu).



Łączenie elementów

Gdy umieścisz już kilka elementów na mapie, czas zacząć je łączyć. Aby połączyć dwa elementy, musisz najpierw je zaznaczyć. Po zaznaczeniu elementów kliknij *Dodaj* obok opcji Link.

Po utworzeniu linku pojedynczy formularz elementu zawiera teraz dodatkową sekcję *Links*. Kliknij *Edytuj*, aby edytować atrybuty linku.



Atrybuty linku:

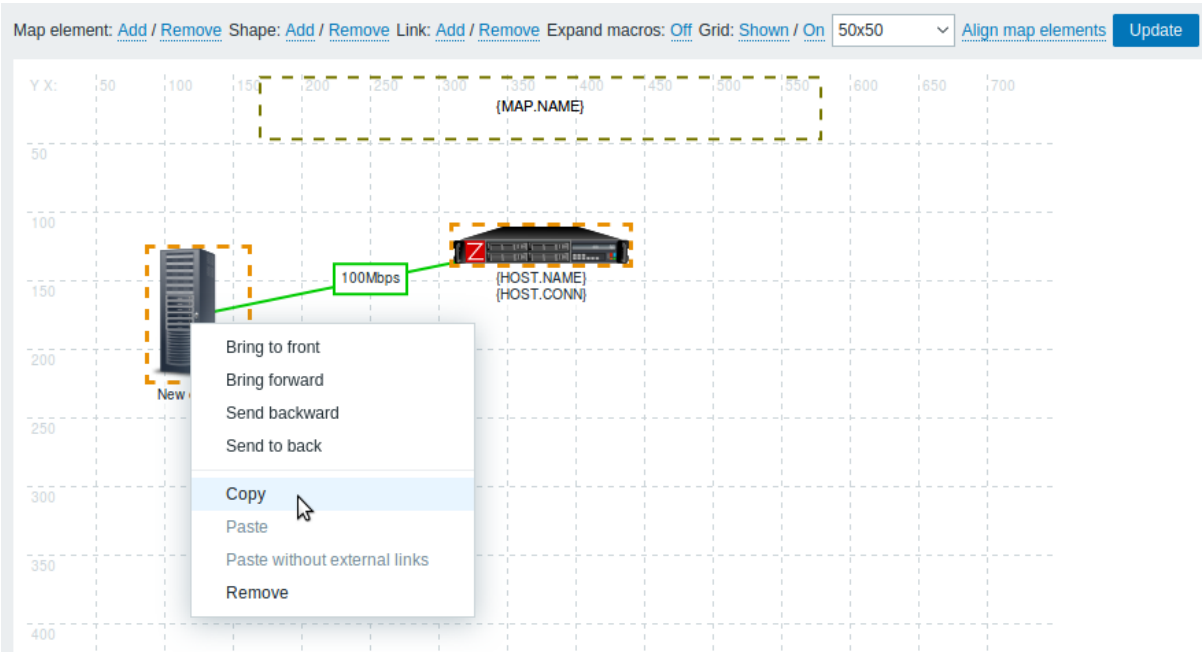
Parameter	Description
Label	Etykieta, która będzie wyświetlana nad linkiem. W tym polu obsługiwane są wyrażenia macros, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`).

Parameter	Description
Show label	Wybierz sposób wyświetlania etykiety linku: **Default** - użyj ustawień *Show map element labels* z konfiguracji mapy (domyślnie); **Always** - zawsze wyświetlane (domyślnie); **Auto hide** - ukryj etykietę, jeśli nie jest najechana kursorem ani zaznaczona.
Connect to	Element, do którego łączy link.
Type (OK)	Domyślny styl linku. Wybierz styl: *Line*, *Bold line*, *Dot* lub *Dashed line*.
Color (OK)	Domyślny kolor linku. Kliknij próbkę koloru, aby wybrać inny kolor.
Indicator type	Wybierz typ wskaźnika linku: **Static link** - bez wskaźników; **Trigger** - umożliwia wskaźniki linku oparte na wyzwalaczach; **Item value** - umożliwia wskaźniki linku oparte na wartości pozycji.
Item	Wybierz pozycję. Wartość tej pozycji będzie wpływać na styl linku. To pole jest dostępne, gdy dla *Indicator type* wybrano *Item value*.
Indicators	Lista link indicators. To pole jest dostępne, gdy *Indicator type* jest ustawiony na *Trigger* lub *Item value*. Jeśli ustawiono *Trigger*, wyzwalacze mogą być przypisywane jako wskaźniki linku. Gdy wyzwalacz znajduje się w stanie *Problem*, jego styl (wybrany kolor i typ linii) zostanie zastosowany do linku. Jeśli ustawiono *Item value* i w polu *Item* wybrano pozycję, wartości pozycji mogą być przypisywane jako wskaźniki linku. Gdy wartość pozycji osiągnie określony próg (dla pozycji numerycznej) lub dopasuje wzorzec (dla pozycji tekstowej), jej styl (wybrany kolor i typ linii) zostanie zastosowany do linku.

Przenoszenie i kopiowanie elementów

Kilka zaznaczonych elementów można **przenieść** w inne miejsce na mapie, klikając jeden z zaznaczonych elementów, przytrzymując przycisk myszy i przesuwając kursor do żądanej lokalizacji.

Jeden lub więcej elementów można **skopiować**, zaznaczając elementy, a następnie klikając zaznaczony element prawym przyciskiem myszy i wybierając *Copy* z menu.

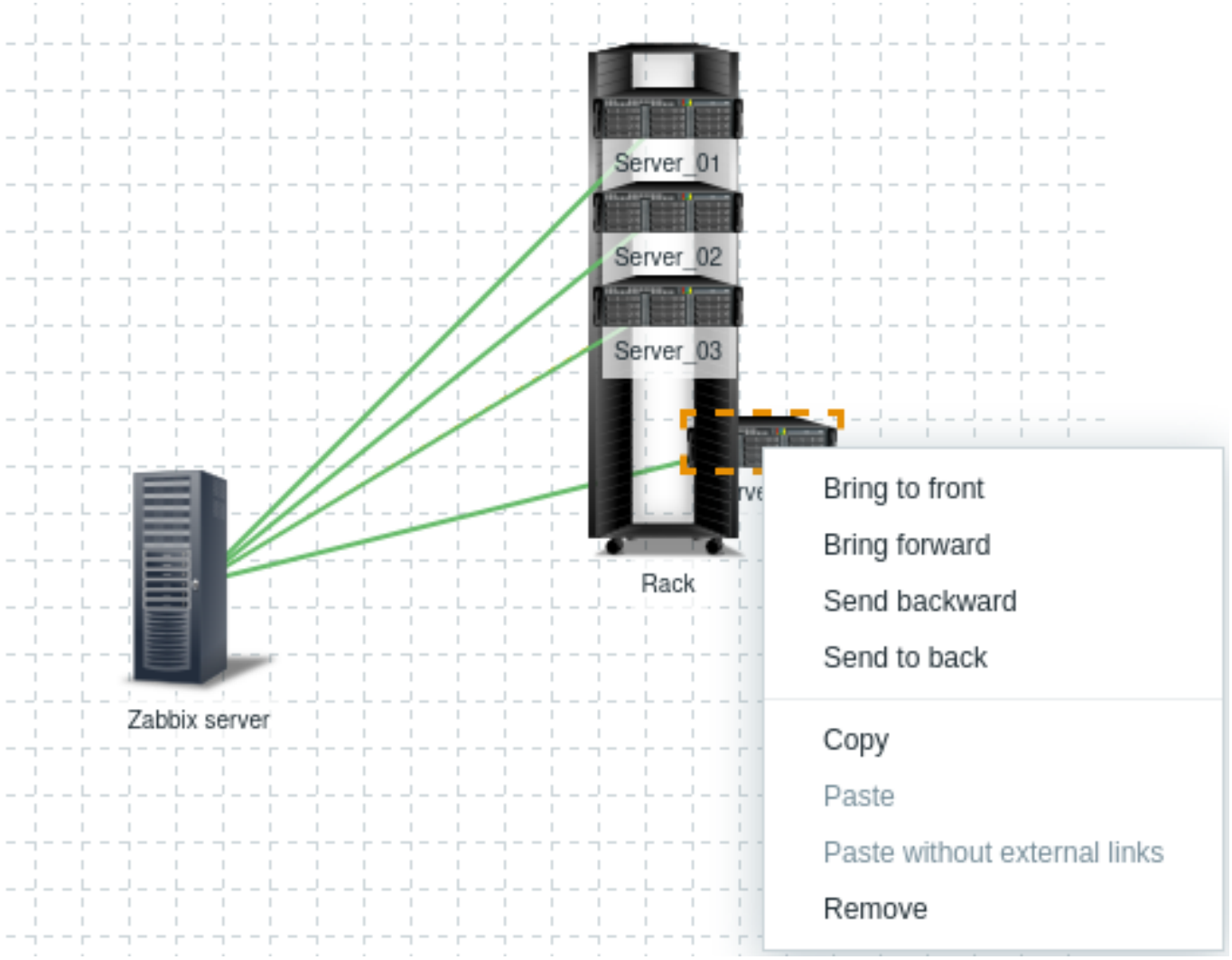


Aby wkleić elementy, kliknij obszar mapy prawym przyciskiem myszy i wybierz *Paste* z menu. Opcja *Paste without external links* wklei elementy, zachowując tylko połączenia między zaznaczonymi elementami.

Kopiowanie i wklejanie działa w obrębie tego samego okna przeglądarki. Skróty klawiaturowe nie są obsługiwane.

Porządkowanie elementów

Aby przenieść jeden element przed drugi (lub odwrotnie), kliknij element prawym przyciskiem myszy i wybierz *Przesuń do przodu/Przenieś na wierzch* lub *Przesuń do tyłu/Przenieś na spód*.

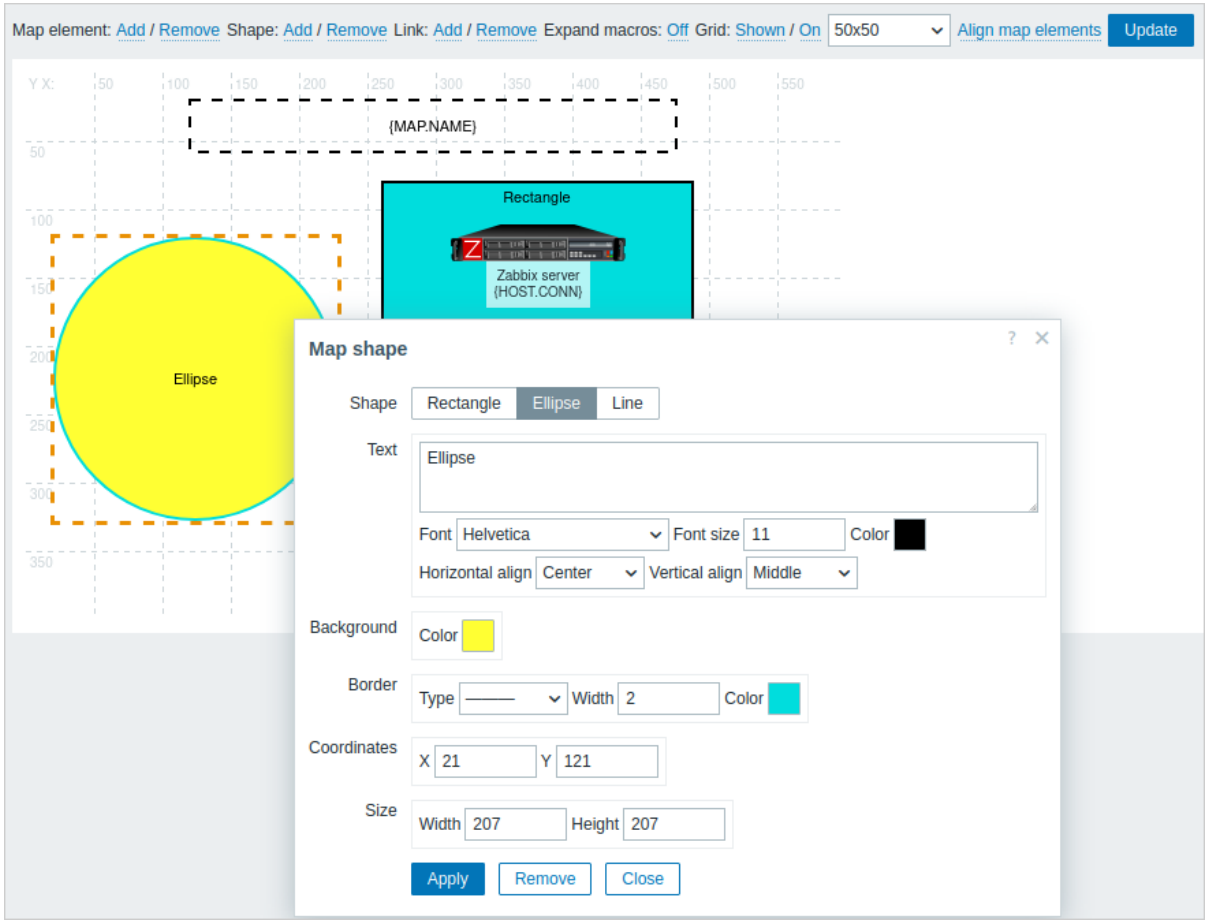


Dodawanie kształtów

Oprócz elementów mapy można również dodawać niektóre kształty. Kształty nie są elementami mapy; stanowią jedynie reprezentację wizualną. Na przykład kształt prostokąta może służyć jako tło do grupowania niektórych hostów. Można dodawać kształty prostokąta i elipsy.

Aby dodać kształt, kliknij *Dodaj* obok Shape. Nowy kształt pojawi się w lewym górnym rogu mapy. Przeciągnij go i upuść w dowolnym miejscu.

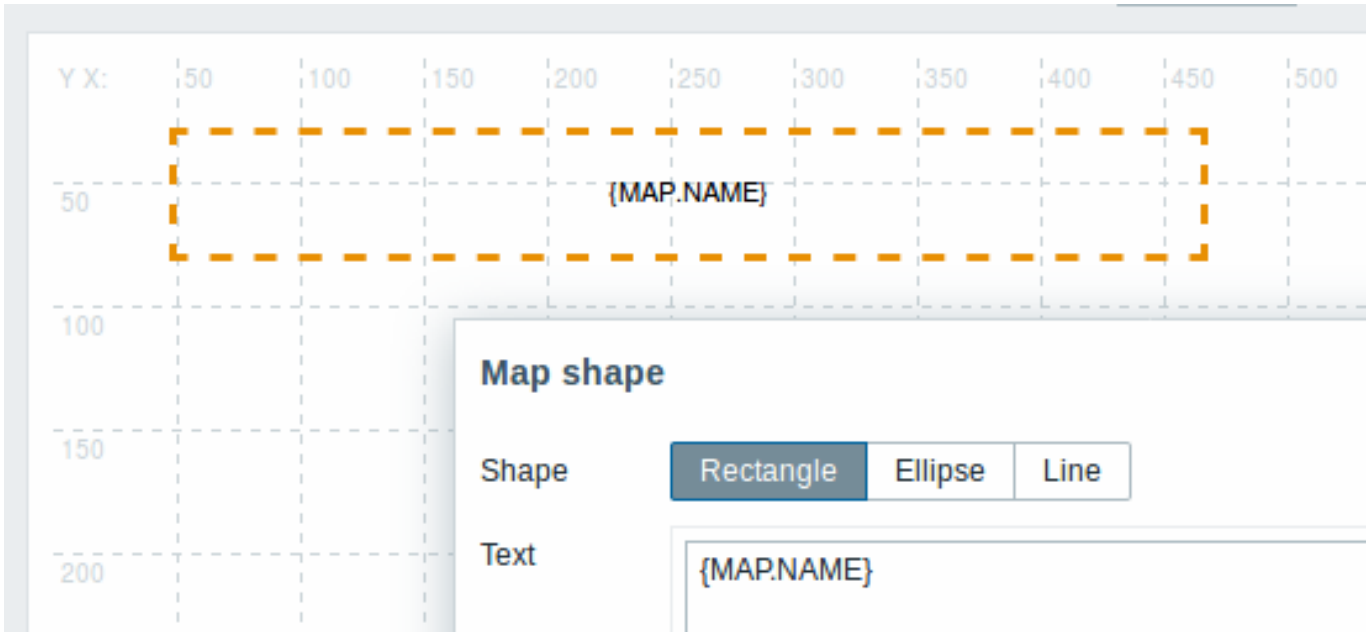
Nowy kształt jest dodawany z domyślnymi kolorami. Po kliknięciu kształtu wyświetlany jest formularz, w którym można dostosować wygląd kształtu, dodać tekst itd.



Aby zaznaczyć kształty, zaznacz jeden z nich, a następnie przytrzymaj *Ctrl*, aby zaznaczyć pozostałe. Po zaznaczeniu kilku kształtów można zbiorczo aktualizować wspólne właściwości, podobnie jak w przypadku elementów.

W kształtach można dodawać tekst. W tekście obsługiwane są makra wyrażeń, ale tylko z funkcjami `avg`, `last`, `min` i `max`, z czasem jako parametrem (na przykład `{?avg(/host/key,1h)}`).

Aby wyświetlać tylko tekst, można ukryć kształt, usuwając jego obramowanie (wybierz 'None' w polu *Border*). Na przykład zwróć uwagę, że makro {MAP.NAME}, widoczne na powyższym zrzucie ekranu, jest w rzeczywistości kształtem prostokąta z tekstem, który można zobaczyć po kliknięciu makra:



{MAP.NAME} jest rozwijane do skonfigurowanej nazwy mapy podczas wyświetlania mapy.

Jeśli w tekście użyto hiperłączy, stają się one klikalne podczas wyświetlania mapy.

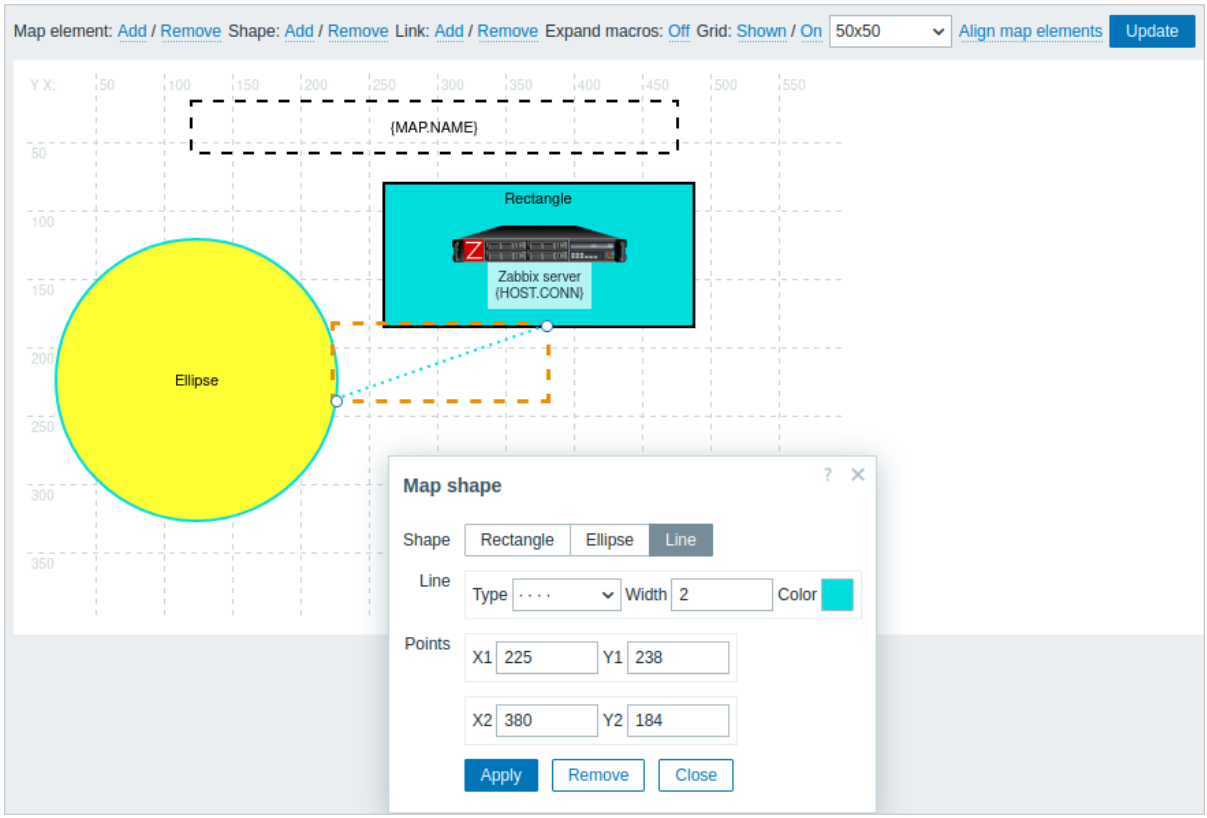
Zawijanie tekstu jest zawsze włączone wewnątrz kształtów. Jednak w obrębie elipsy linie są zawijane tak, jakby elipsa była prostokątem. Zawijanie wyrazów nie jest zaimplementowane, więc długie słowa (słowa, które nie mieszczą się w kształcie) nie są

zawijane, lecz maskowane (na stronie edycji mapy) lub przycinane (na innych stronach z mapami).

Dodawanie linii

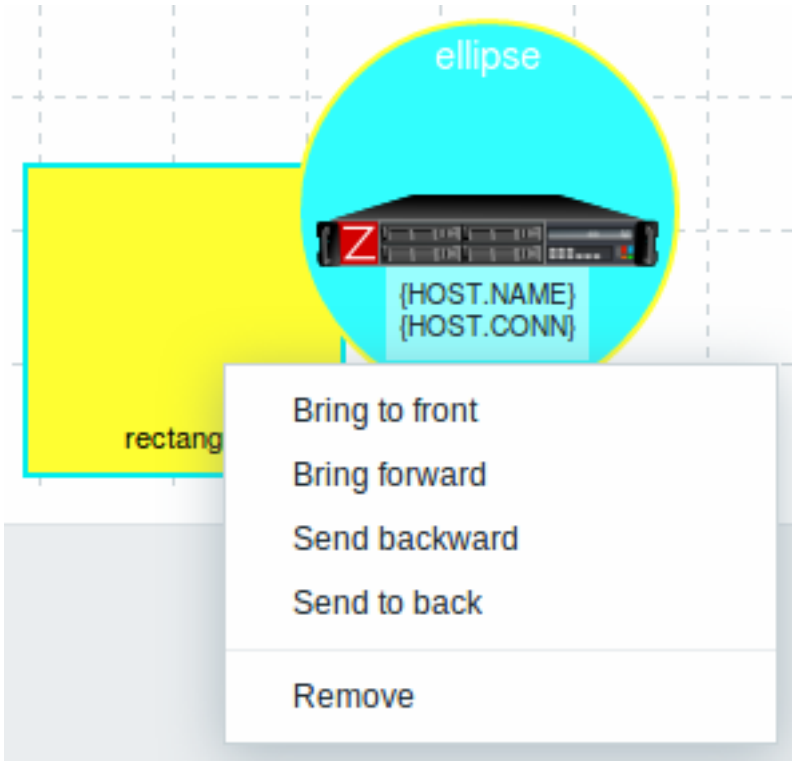
Oprócz kształtów można również dodawać linie. Linie mogą być używane do łączenia elementów lub kształtów na mapie.

Aby dodać linię, kliknij *Add* obok Shape. Nowy kształt pojawi się w lewym górnym rogu mapy. Zaznacz go i kliknij *Line* w formularzu edycji, aby zmienić kształt w linię. Następnie dostosuj właściwości linii, takie jak typ linii, szerokość, kolor itp.



Porządkowanie kształtów i linii

Aby przenieść jeden kształt przed drugi (lub odwrotnie), kliknij kształt prawym przyciskiem myszy, aby wyświetlić menu kształtu mapy.

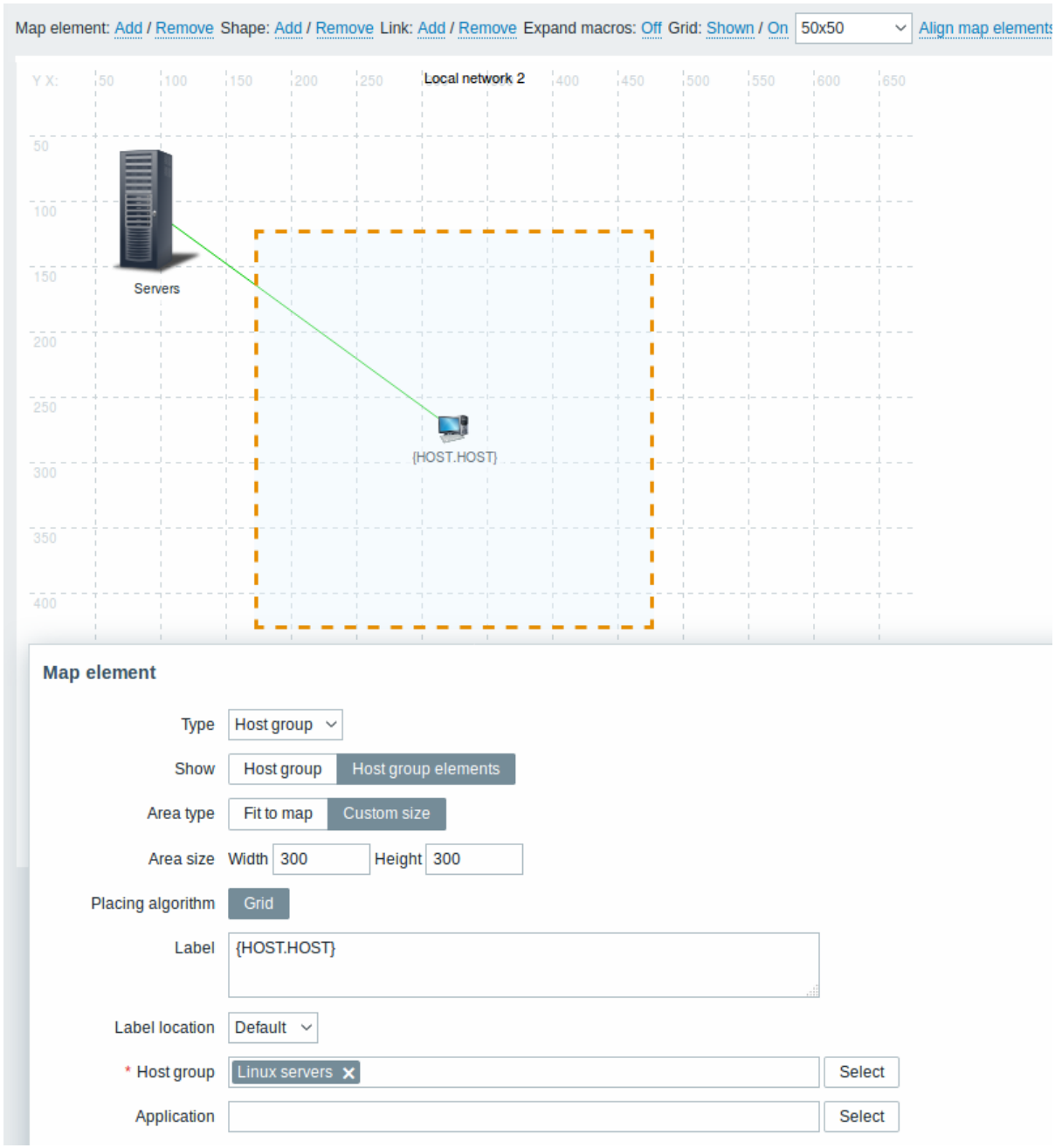


Elementy grupy hostów

Przegląd

Ta sekcja wyjaśnia, jak dodać element typu *Grupa hostów* podczas konfigurowania mapy sieci.

Konfiguracja



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

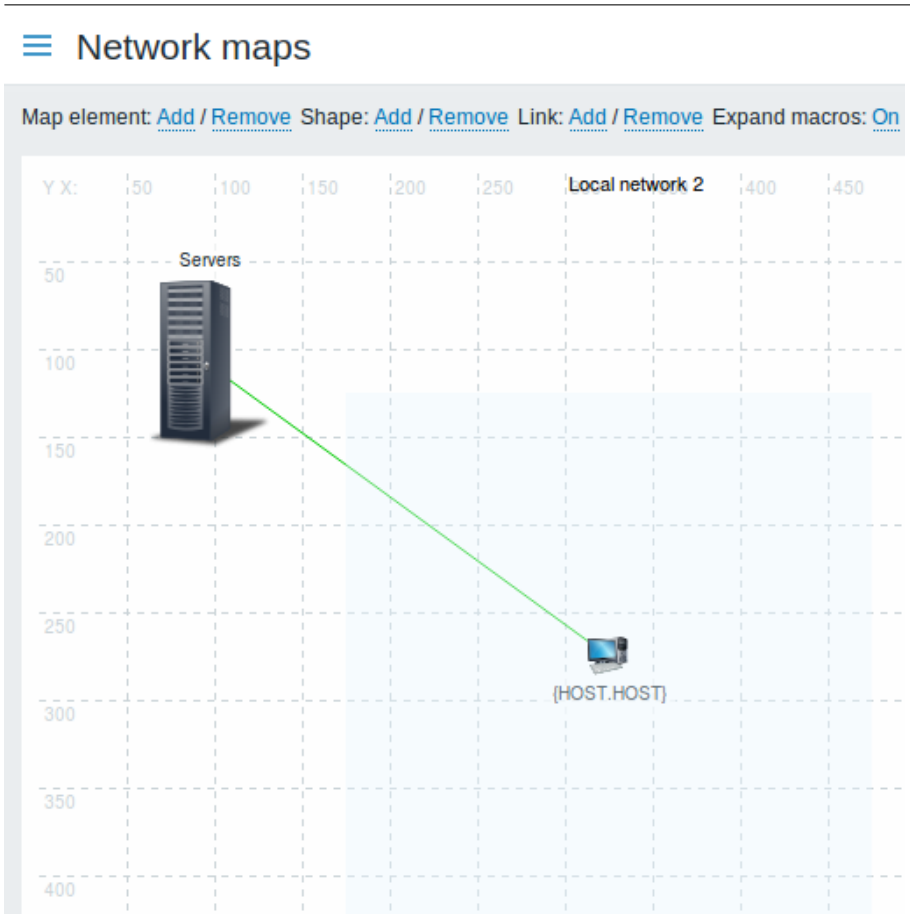
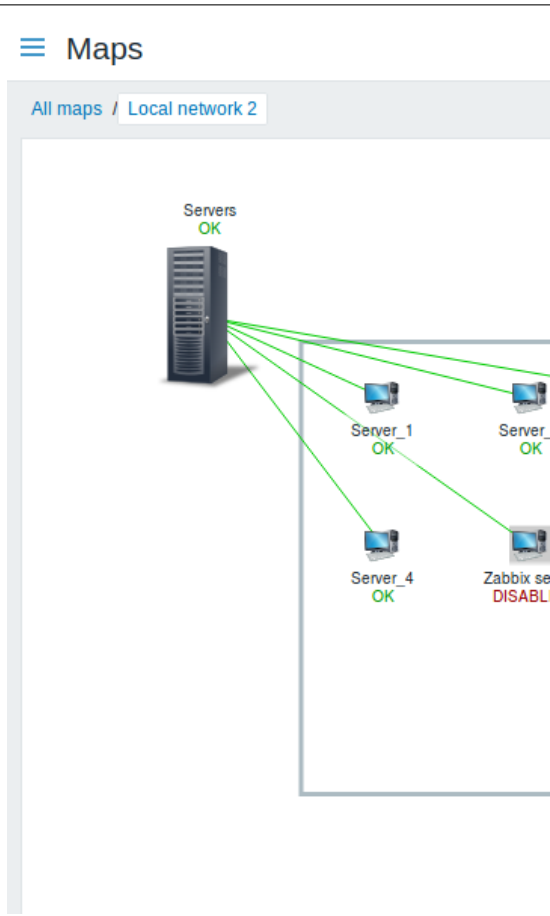
Ta tabela zawiera parametry typowe dla typu elementu *Host group*:

Parametr	Opis
Type	Wybierz typ elementu: **Host group** - Ikona przedstawiająca stan wszystkich wyzwalaczy wszystkich hostów należących do wybranej grupy.
Show	Opcje wyświetlania: **Host group** - Wybranie tej opcji spowoduje wyświetlenie jednej ikony zawierającej odpowiednie informacje o określonej grupie hostów. **Host group elements** - Wybranie tej opcji spowoduje wyświetlenie wielu ikon zawierających odpowiednie informacje o każdym elemencie (hoście) określonej grupy hostów.

Parametr	Opis
Area type	To ustawienie jest dostępne, jeśli dla opcji *Show* wybrano wartość *Host group elements*: **Fit to map** - Wszystkie elementy grupy hostów są rozmieszczane równomiernie na mapie. **Custom size** - Ręczne ustawienie obszaru mapy dla wszystkich wyświetlanych elementów grupy hostów.
Area size	To ustawienie jest dostępne, jeśli dla opcji *Area type* wybrano wartość *Custom size*: **Width** - Wartość w pikselach określająca szerokość obszaru mapy. **Height** - Wartość w pikselach określająca wysokość obszaru mapy.
Placing algorithm	**Grid** - Jedyna dostępna opcja wyświetlania wszystkich elementów grupy hostów.
Label	Etykieta ikony, dowolny ciąg znaków. W etykietach można używać makr i ciągów wielowierszowych. Jeśli typ elementu mapy to *Host group*, określenie konkretnych makr wpływa na widok mapy, wyświetlając odpowiednie informacje o każdym hoście. Na przykład jeśli użyto makra `{HOST.IP}`, widok edycji mapy będzie wyświetlać tylko samo makro {HOST.IP}, natomiast widok mapy będzie zawierać i wyświetlać unikalny adres IP każdego hosta.

Wyświetlanie elementów grupy hostów

Ta opcja jest dostępna, jeśli wybrano opcję wyświetlania *Elementy grupy hostów*.
Po wybraniu *Elementy grupy hostów* jako opcji *wyświetlania* początkowo zobaczysz tylko jedną ikonę grupy hostów.
Jednak po zapisaniu mapy i przejściu do widoku mapy zobaczysz, że mapa zawiera wszystkie elementy (hosty) określonej grupy hostów:

Widok edycji mapy	Widok mapy
	

Zwróć uwagę, jak używane jest makro `{HOST.HOST}`.
Podczas edycji mapy nazwa makra nie jest rozwinięta, natomiast w widoku mapy wyświetlane są unikatowe nazwy hostów.

Wskaźniki linków

Przegląd

Możesz przypisać wskaźniki do łącza między elementami na mapie sieci.

Wskaźniki mogą być oparte na wyzwalaczach lub wartościach pozycji. Możliwe jest wyświetlanie różnych stylów i kolorów łącza:

- gdy wyzwalacze przechodzą w stan problemu;
- gdy wartość pozycji:
  - osiąga próg (dla pozycji numerycznych);
  - pasuje do wyrażenia regularnego (dla pozycji tekstowych).

Podczas konfigurowania łącza ustawiasz domyślny typ i kolor łącza. Przypisanie wskaźników do łącza pozwala uzależnić styl i kolor łącza od stanu wyzwalacza lub wartości pozycji.

Na przykład, jeśli którykolwiek z przypisanych wyzwalaczy przejdzie w stan problemu, styl i kolor łącza zmienią się, aby to odzwierciedlić. Być może domyślnie łącze było zieloną linią. Teraz, gdy wyzwalacz jest w stanie problemu, łącze może stać się pogrubioną czerwoną linią (jeśli tak je zdefiniowano).

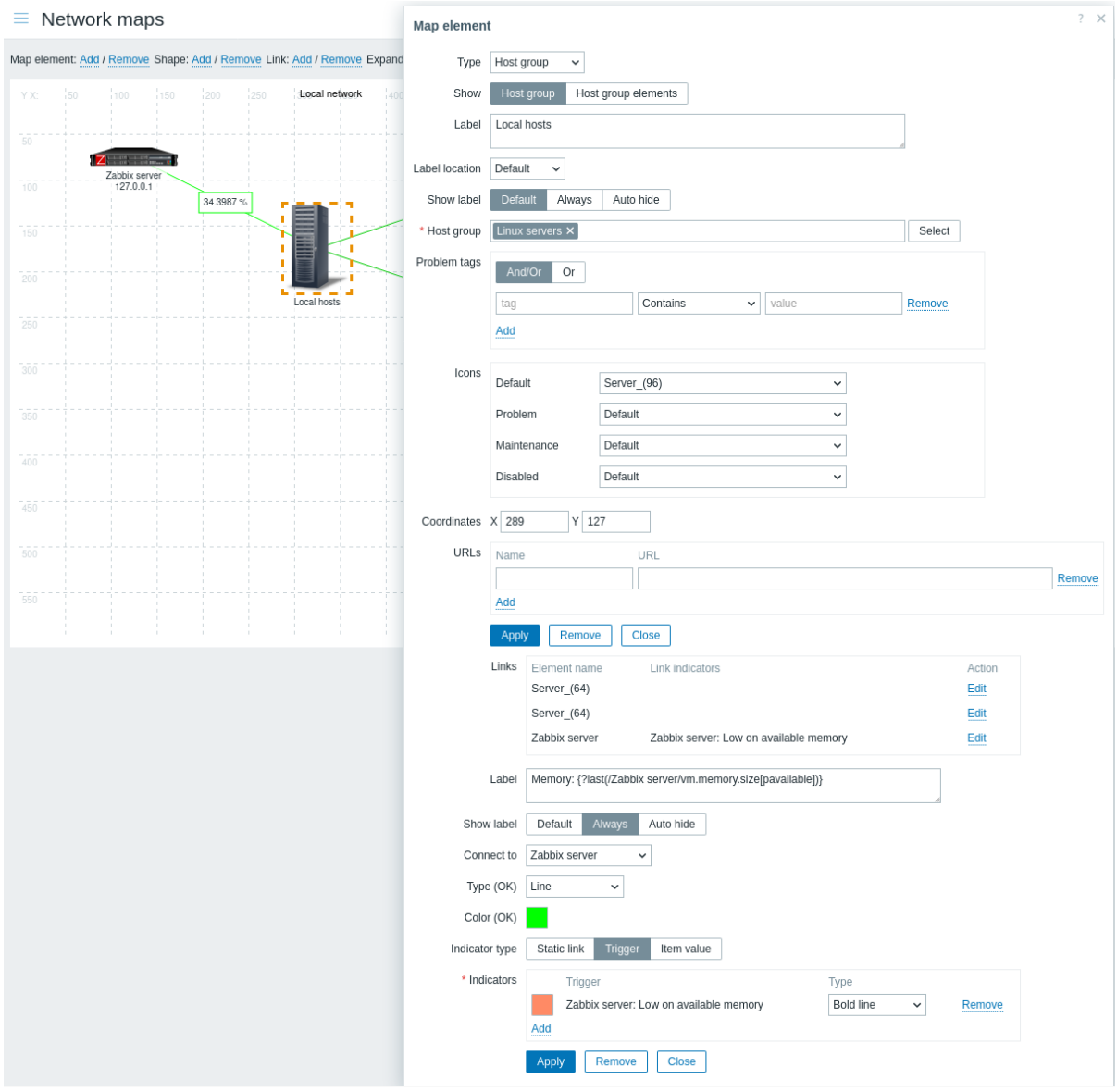
Podobnie, jeśli wartość pozycji osiągnie określony próg lub będzie pasować do określonego wyrażenia regularnego, styl łącza może to odzwierciedlać.

Konfiguracja

Wyzwalacze

Aby przypisać wyzwalacze jako wskaźniki łącza, wykonaj następujące czynności:

- wybierz element mapy
- kliknij *Edit* w sekcji *Links* dla odpowiedniego łącza
- wybierz *Trigger* jako typ wskaźnika
- kliknij *Add* w bloku *Indicators* i wybierz jeden lub więcej wyzwalaczy

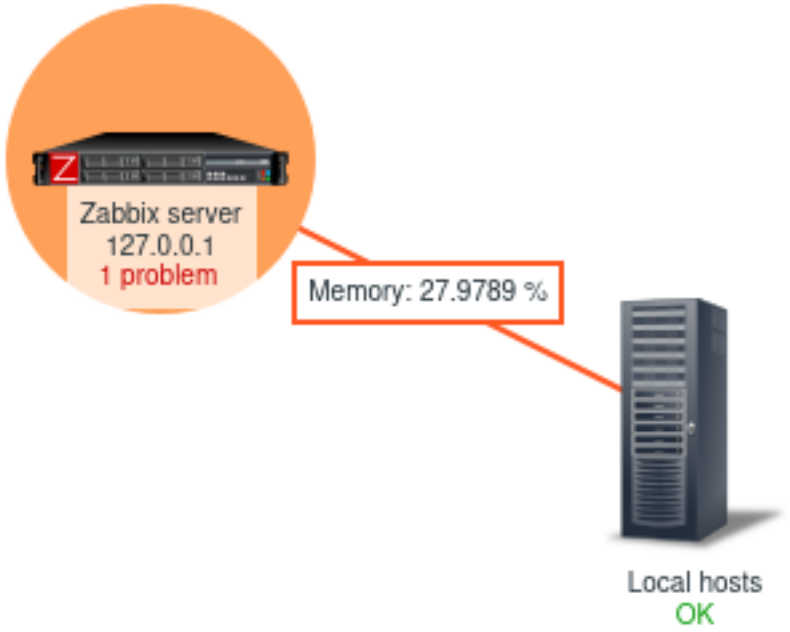


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Dodane wyzwalacze można zobaczyć na liście *Indicators*.

Możesz ustawić typ łącza i kolor dla każdego wskaźnika bezpośrednio z listy. Po zakończeniu kliknij *Apply*, zamknij formularz i kliknij *Update*, aby zapisać zmiany mapy.

W *Monitoring* > *Maps* odpowiedni typ łącza i kolor będą wyświetlane, jeśli **trigger** przejdzie w stan problemu.



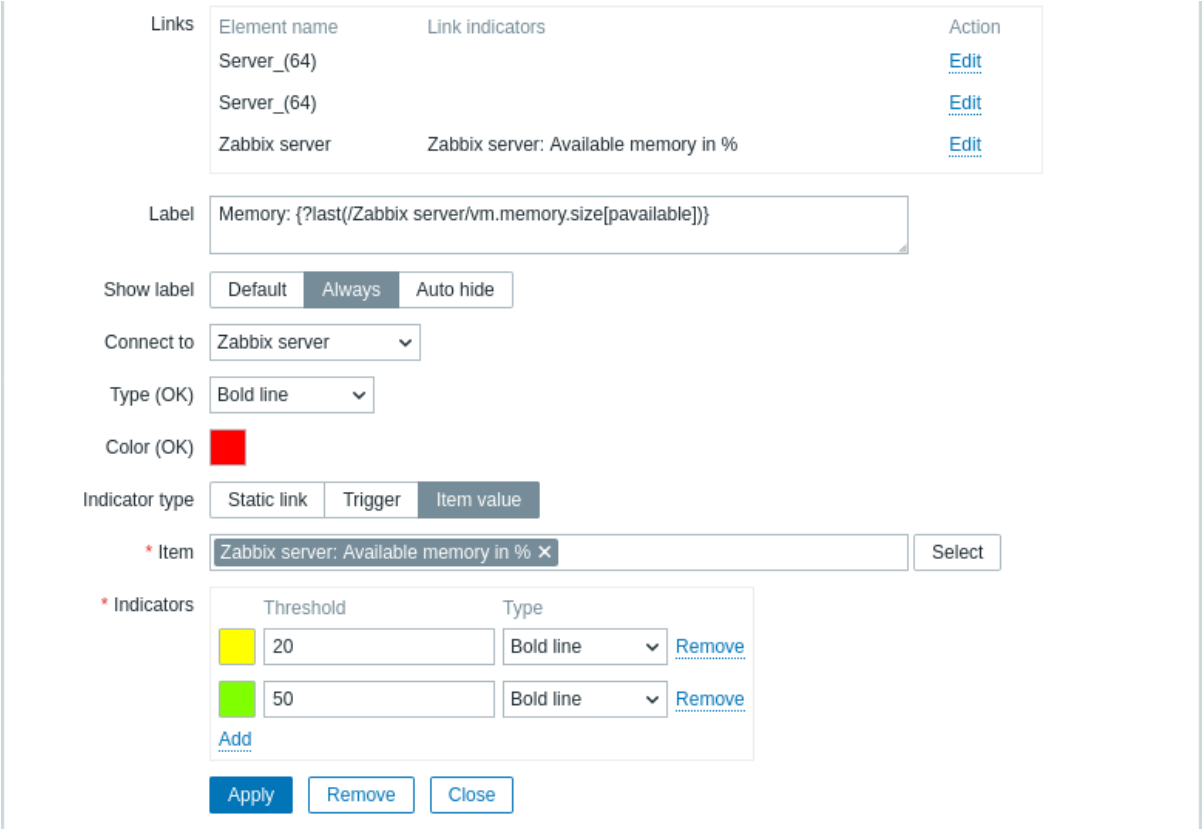
Note:
Jeśli wiele wyzwalaczy przejdzie w stan problemu, problem o najwyższym poziomie ważności zadecyduje o stylu i kolorze łącza. Jeśli do tego samego łącza mapy przypisano wiele wyzwalaczy o tym samym poziomie ważności, pierwszeństwo ma ten o najniższym ID. Należy również pamiętać, że:

1. Ustawienia *Minimum trigger severity* i *Show suppressed problem* z konfiguracji mapy wpływają na to, które problemy są brane pod uwagę.
2. W przypadku wyzwalaczy z wieloma problemami (wielokrotne generowanie problemów), każdy problem może mieć poziom ważności różniący się od poziomu ważności wyzwalacza (zmieniony ręcznie), może mieć inne tagi (z powodu makr) i może być wyciszony.

Wartości pozycji

Aby przypisać wartości pozycji jako wskaźniki łącza, wykonaj następujące czynności:

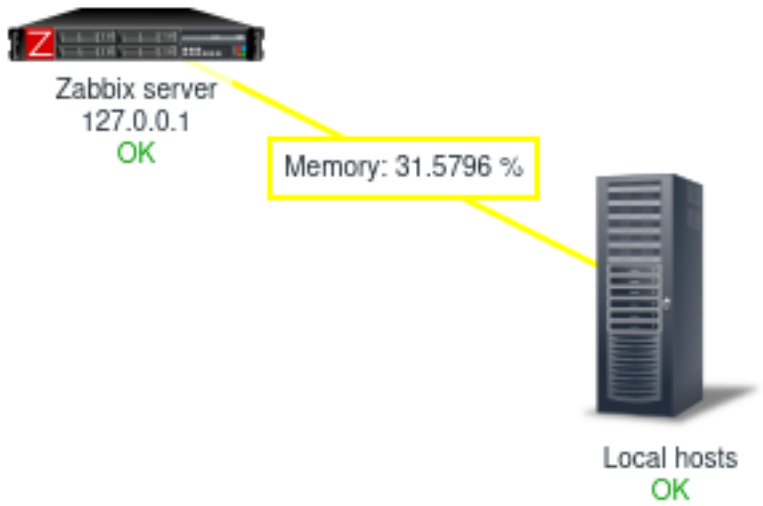
- wybierz element mapy
- kliknij *Edit* w sekcji *Links* dla odpowiedniego łącza
- wybierz *Item value* jako typ wskaźnika
- wybierz pozycję
- dodaj jeden lub więcej progów wartości pozycji albo wzorców w bloku *Indicators*



Dodane progi/wzorce pozycji można zobaczyć na liście *Indicators*.

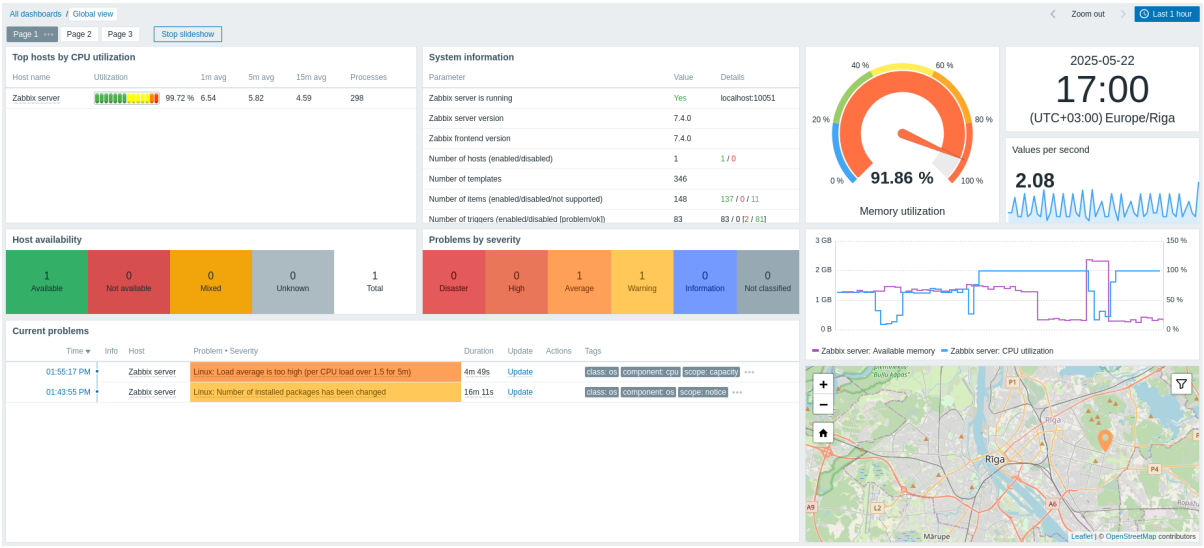
Możesz ustawić typ łącza i kolor dla każdego wskaźnika bezpośrednio z listy. Po zakończeniu kliknij *Apply*, zamknij formularz i kliknij *Update*, aby zapisać zmiany mapy.

W *Monitoring* > *Maps* odpowiedni typ łącza i kolor będą wyświetlane, jeśli **wartość pozycji** osiągnie ustawiony próg (dla numerycznych typów danych) lub dopasuje się do wzorca wyrażenia regularnego (dla tekstowych typów danych).



Pulpity

Pulpity - zarówno globalne pulpity, jak i pulpity hosta - zapewniają rozbudowaną platformę wizualizacji z takimi widgetami i narzędziami jak nowoczesne wykresy, mapy, pokazy slajdów i inne.



Szablony i grupy szablonów

Przegląd

Użycie szablonów to doskonały sposób na zmniejszenie nakładu pracy i usprawnienie konfiguracji Zabbix. Szablon to zestaw encji, który można wygodnie zastosować do wielu hostów.

Encje te mogą obejmować:

- pozycje
- wyzwalacze
- wykresy
- pulpity
- reguły wykrywania niskiego poziomu
- scenariusze WWW

Ponieważ wiele hostów w rzeczywistości jest identycznych lub dość podobnych, naturalne jest, że zestaw encji (pozycji, wyzwalaczy, wykresów, ...) utworzony dla jednego hosta może być przydatny dla wielu innych. Oczywiście można skopiować je do każdego nowego hosta, ale wymagałoby to dużo ręcznej pracy. Zamiast tego, korzystając z szablonów, można skopiować je do jednego szablonu, a następnie zastosować ten szablon do dowolnej liczby hostów.

Gdy szablon jest podłączony do hosta, wszystkie encje (pozycje, wyzwalacze, wykresy, ...) ze szablonu są dodawane do hosta. Szablony są przypisywane bezpośrednio do każdego pojedynczego hosta (a nie do grupy hostów).

Szablony są często używane do grupowania encji dla określonych usług lub aplikacji (takich jak Apache, MySQL, PostgreSQL, Postfix...) i następnie stosowane do hostów, na których działają te usługi.

Kolejną korzyścią z używania szablonów jest możliwość wprowadzania zmian dla wszystkich hostów jednocześnie. Jednorazowa zmiana na poziomie szablonu zostanie propagowana do wszystkich powiązanych hostów.

Szablony są zorganizowane w grupy szablonów.

Przejdź do tworzenia i konfigurowania szablonu.

Szablony dostępne od razu po instalacji

Przegląd

Zabbix udostępnia coraz większy zestaw wstępnie skonfigurowanych szablonów, aby uprościć i przyspieszyć konfigurację monitorowanych celów.

Wszystkie szablony dostępne od razu po instalacji znajdują się w sekcji *Zbieranie danych* > *Szablony*.

Podczas aktualizacji Zabbix szablony nie są aktualizowane automatycznie. Zapobiega to nadpisaniu wprowadzonych przez Ciebie niestandardowych zmian. Aby ręcznie zaktualizować lub dodać szablony, zobacz poniższą sekcję *Aktualizacja szablonów*.

Użyj paska bocznego, aby uzyskać dostęp do informacji o określonych typach szablonów i wymaganiach dotyczących ich działania.

Globalne pulpity zaprojektowane do współpracy z istniejącymi szablonami Zabbix można pobrać z repozytorium pulpitów i zaimportować do środowiska Zabbix. Szczegółowy opis każdego pulpitu, w tym wymagania wstępne, znajduje się w pliku `README` danego pulpitu.

Zobacz także:

- Łączenie szablonu
- Znane problemy dotyczące szablonów

Aktualizacja szablonów

Aby zaktualizować lub dodać **pojedynczy szablon**:

1. Otwórz katalog `templates` w repozytorium Git Zabbix.
2. Wybierz gałąź (w sekcji *Source*) odpowiadającą używanej wersji Zabbix (np. *master* dla wersji rozwojowej).
3. Znajdź szablon, który chcesz dodać lub zaktualizować (np. MySQL by Zabbix agent).
4. Skopiuj zawartość pliku szablonu (np. `template_db_mysql_agent.yaml`) i zapisz ją w pliku lokalnym.
5. W interfejsie użytkownika Zabbix przejdź do *Zbieranie danych* > *Szablony* i zaimportuj plik lokalny.

> **Note:**
> Typy nośników webhook można aktualizować w podobny sposób: pobierz je z katalogu `media` i zaimportuj w sekcji *Alerty* > *Typy nośników*.

Aby zaktualizować lub dodać **wiele szablonów**:

1. Pobierz archiwum pakietów szablonów odpowiednie dla używanego serwera WWW:
   - Nginx - `templates_bundles-8.0-1M-nginx.zip`
   - Apache - `templates_bundles-8.0-2M-apache.zip`
2. Wypakuj archiwum. Zawiera ono pliki YAML, z których każdy zawiera wiele szablonów, oraz plik indeksu zawierający listę szablonów znajdujących się w poszczególnych plikach YAML.
3. W interfejsie użytkownika Zabbix przejdź do *Zbieranie danych* > *Szablony* i zaimportuj pliki YAML pojedynczo.

> **Note:**
> Typy nośników webhook można aktualizować w podobny sposób: pobierz plik `media_bundle-8.0-000.yaml` z archiwum pakietów szablonów i zaimportuj go w sekcji *Alerty* > *Typy nośników*.

Aby zaktualizować lub dodać **wszystkie szablony** jednocześnie, możesz użyć skryptu `import_templates.sh` dołączonego do archiwum pakietów szablonów. Skrypt wymaga narzędzi `curl` i `jq` oraz korzysta z metody API Zabbix `configuration.import`. Podczas uruchamiania skryptu podaj katalog zawierający pliki YAML, a także adres URL API Zabbix i token uwierzytelniający.

> **Warning:**
> Skrypt `import_templates.sh` nadpisuje wszystkie istniejące szablony bez możliwości przejrzenia lub cofnięcia zmian w szablonach.

Przykład:

```
/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
```

Zgodność szablonu z Host Wizard

Host Wizard jest zgodny ze wszystkimi szablonami dostępnymi od razu po instalacji.

Możesz również dostosować własne szablony do współpracy z Host Wizard:

1. W frontend Zabbix przejdź do *Zbieranie danych* > *Szablony* i wyeksportuj własny szablon.
2. Edytuj wyeksportowany plik:
   - Dodaj element `wizard_ready` szablonu, aby wskazać zgodność z Host Wizard.
   - Opcjonalnie dodaj element `readme` lub `config` szablonu, zgodnie z potrzebami.
3. W frontend Zabbix przejdź do *Zbieranie danych* > *Szablony* i zaimportuj zaktualizowany plik szablonu.

Działanie szablonu Zabbix agent

Kroki zapewniające prawidłowe działanie szablonów, które zbierają metryki za pomocą Zabbix agent:

1. Upewnij się, że Zabbix agent jest zainstalowany na hoście. W przypadku kontroli aktywnych upewnij się również, że adres serwera/proxy Zabbix został dodany do parametru „ServerActive" w pliku konfiguracyjnym agenta.
2. Połącz szablon z docelowym hostem (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne jego wcześniejsze zaimportowanie).
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby zezwalała na udostępnianie danych Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- 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

Operacja szablonu Zabbix agent 2

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki za pomocą Zabbix agent 2:

1. Upewnij się, że agent 2 jest zainstalowany na host, a zainstalowana wersja zawiera wymagany plugin. W niektórych przypadkach może być konieczne wcześniejsze uaktualnienie agent 2.
2. Powiąż szablon z docelowym host (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).
3. W razie potrzeby dostosuj wartości makr szablonu. Zwróć uwagę, że makra użytkownika mogą służyć do nadpisywania parametrów konfiguracji.
4. Skonfiguruj monitorowaną instancję tak, aby zezwalała na udostępnianie danych z Zabbix.

Attention:
Szablony Zabbix agent 2 działają w połączeniu z pluginami. Podczas gdy podstawową konfigurację można wykonać, po prostu dostosowując makra użytkownika, głębszą personalizację można osiągnąć, samodzielnie konfigurując plugin. Na przykład, jeśli plugin obsługuje nazwane sesje, możliwe jest monitorowanie kilku encji tego samego rodzaju (np. MySQL1 i MySQL2) przez określenie nazwanej sesji z własnym URI, nazwą użytkownika i hasłem dla każdej encji w pliku konfiguracyjnym.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- 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

Operacja szablonu HTTP

Wykonaj poniższe kroki, aby zapewnić prawidłowe działanie szablonów zbierających metryki za pomocą agenta HTTP:

1. Utwórz host w Zabbix odpowiadający monitorowanemu celowi. Jako główny interfejs hosta określ adres IP lub nazwę DNS monitorowanego celu. Jest to wymagane, aby makro `{HOST.CONN}` było prawidłowo rozwijane w pozycjach szablonu.
2. Połącz szablon z hostem utworzonym w kroku 1 (jeśli szablon nie jest dostępny w używanej instalacji Zabbix, może być konieczne jego wcześniejsze zaimportowanie).
3. W razie potrzeby dostosuj wartości makr na poziomie szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby zezwalała na udostępnianie danych systemowi Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku `README` szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- 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 są przestarzałe i zostały zastąpione przez szablony *Kubernetes Cluster by HTTP* oraz *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

Operacja szablonu IPMI

Szablony IPMI nie wymagają żadnej specjalnej konfiguracji. Aby rozpocząć monitorowanie, połącz szablon z docelowym hostem (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępny szablon:

- Chassis by IPMI

Operacja szablonu JMX

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki przez JMX:

1. Upewnij się, że Zabbix Java gateway jest zainstalowany i poprawnie skonfigurowany.
2. Połącz szablon z docelowym hostem. Host powinien mieć skonfigurowany interfejs JMX. Jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu.
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby umożliwiała udostępnianie danych z Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- 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

Operacja szablonu ODBC

Kroki zapewniające poprawne działanie szablonów, które zbierają metryki za pomocą monitorowania ODBC:

1. Upewnij się, że wymagany sterownik ODBC jest zainstalowany na serwerze Zabbix lub proxy.
2. Połącz szablon z docelowym hostem (jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne wcześniejsze zaimportowanie szablonu).
3. W razie potrzeby dostosuj wartości makr szablonu.
4. Skonfiguruj monitorowaną instancję tak, aby umożliwiała udostępnianie danych z Zabbix.

Szczegółowy opis szablonu, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku README szablonu (dostępnym po kliknięciu nazwy szablonu).

Dostępne są następujące szablony:

- MariaDB by ODBC
- MSSQL by ODBC
- MySQL by ODBC
- Oracle by ODBC
- Percona by ODBC
- PostgreSQL by ODBC

Ustandaryzowane szablony dla urządzeń sieciowych

Przegląd

Aby zapewnić monitorowanie urządzeń sieciowych, takich jak przełączniki i routery, utworzyliśmy dwa tak zwane modele: dla samego urządzenia sieciowego (zasadniczo jego obudowy) oraz dla interfejsu sieciowego.

Dostępne są szablony dla wielu rodzin urządzeń sieciowych. Wszystkie szablony obejmują (tam, gdzie możliwe jest pobranie tych danych z urządzenia):

- Monitorowanie awarii obudowy (zasilacze, wentylatory i temperatura, ogólny stan)
- Monitorowanie wydajności obudowy (pozycje CPU i pamięci)
- Zbieranie informacji o obudowie (numery seryjne, nazwa modelu, wersja oprogramowania układowego)
- Monitorowanie interfejsu sieciowego z użyciem IF-MIB i EtherLike-MIB (stan interfejsu, obciążenie ruchem interfejsu, stan dupleksu dla Ethernet)

Jeśli szablon nie jest dostępny w Twojej instalacji Zabbix, może być konieczne najpierw zaimportowanie szablonu.

Jeśli importujesz nowe szablony dostarczane domyślnie, możesz również zaktualizować globalne wyrażenie regularne `@Network interfaces for discovery` do:

```
Does not match regular expression: ^Software Loopback Interface
Does not match regular expression: ^(In)?[lL]oop[bB]ack[0-9._]*$
Does not match regular expression: ^NULL[0-9.]*$
Does not match regular expression: ^[lL]o[0-9.]*$
Does not match regular expression: ^[sS]ystem$
Does not match regular expression: ^Nu[0-9.]*$
```

aby odfiltrować interfejsy loopback i interfejsy null w większości systemów.

Urządzenia

Lista rodzin urządzeń, dla których dostępne są szablony:

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Alcatel Timetra TiMOS SNMP	Alcatel	Alcatel Timetra	ALCATEL SR 7750	TiMOS	TIMETRA-SYSTEM-MIB,TIMETRA-CHASSIS-MIB	Certified
Aruba CX 8300s by SNMP	HPE Aruba Networking	Seria przełączników 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	Certified
Brocade FC SNMP	Brocade	Przełączniki Brocade FC	Brocade 300 SAN Switch-	-	SW-MIB,ENTITY-MIB	Performance Fault
Brocade_Foundry Stackable SNMP	Brocade	Brocade ICX	Brocade ICX6610, Brocade ICX7250-48, Brocade ICX7450-48F		FOUNDRY-SN-AGENT-MIB, FOUNDRY-SN-STACKING-MIB	Certified
Brocade_Foundry Nonstackable SNMP	Brocade, Foundry	Brocade MLX, Foundry	Brocade MLXe, Foundry FLS648, Foundry FWSX424		FOUNDRY-SN-AGENT-MIB	Performance Fault
Check Point Next Generation Firewall by SNMP	Check Point	Zapora nowej generacji	-	Gaia	HOST-RESOURCES-MIB, CHECKPOINT-MIB, UCD-SNMP-MIB, SNMPv2-MIB, IF-MIB	Certified
Ciena 3906 by SNMP	Ciena	Ciena CPE	Ciena 3906	SAOS	WWP-LEOS-BLADE-MIB, WWP-LEOS-CHASSIS-MIB, WWP-LEOS-SYSTEM-CONFIG-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
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	Certified
Cisco IOS SNMP	Cisco	Cisco IOS ver > 12.2 3.5	Cisco C2950	IOS	CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB,CISCO-ENVMON-MIB	Certified
Cisco IOS versions 12.0_3_T-12.2_3.5 SNMP	Cisco	Cisco IOS > 12.0 3 T i 12.2 3.5	-	IOS	CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB,CISCO-ENVMON-MIB	Certified
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	Certified
D-Link DES_DGS Switch SNMP	D-Link	Przełączniki DES/DGX	D-Link DES-xxxx/DGS-xxxx,DLINK DGS-3420-26SC	-	DLINK-AGENT-MIB,EQUIPMENT-MIB,ENTITY-MIB	Certified
D-Link DES 7200 SNMP	D-Link	DES-7xxx	D-Link DES 7206	-	ENTITY-MIB,MY-SYSTEM-MIB,MY-PROCESS-MIB,MY-MEMORY-MIB	Performance Fault Interfaces
Dell Force S-Series SNMP	Dell	Dell Force S-Series	S4810		F10-S-SERIES-CHASSIS-MIB	Certified
Extreme Exos SNMP	Extreme	Extreme EXOS	X670V-48x	EXOS	EXTREME-SYSTEM-MIB,EXTREME-SOFTWARE-MONITOR-MIB	Certified
FortiGate by SNMP	Fortinet	FortiGate (NGFW)	-	FortiOS	HOST-RESOURCES-MIB FORTINET-FORTIGATE-MIB FORTINET-CORE-MIB SNMPv2-MIB IF-MIB ENTITY-MIB	Performance Inventory
HP Comware HH3C SNMP	HP	HP (H3C) Comware	HP A5500-24G-4SFP HI Switch		HH3C-ENTITY-EXT-MIB,ENTITY-MIB	Certified
HP Enterprise Switch SNMP	HP	Przełącznik HP Enterprise	HP ProCurve J4900B Switch 2626, HP J9728A 2920-48G Switch		STATISTICS-MIB,NETSWITCH-MIB,HP-ICF-CHASSIS,ENTITY-MIB,SEMI-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Huawei AR600 by SNMP	Huawei	Routery serii Huawei NetEngine 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	Certified
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	Certified
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	Certified
Huawei VRP by SNMP	Huawei	Huawei VRP	S2352P-EI		ENTITY-MIB,HUAWEI-ENTITY-EXTENT-MIB	Certified
Intel_Qlogic Infiniband SNMP	Intel/QLogic	Urządzenia Intel/QLogic Infiniband	Infiniband 12300		ICS-CHASSIS-MIB	Fault Inventory
Juniper SNMP	Juniper	Modele MX,SRX,EX	Juniper MX240, Juniper EX4200-24F	JunOS	JUNIPER-MIB	Certified
Juniper MX NETCONF	Juniper	Modele MX	Juniper MX204 Edge Router	JunOS 24.2R1-S1.10	*Nie dotyczy (używa NETCONF przez SSH)*	Certified
Juniper MX SNMP	Juniper	Modele MX	Juniper MX204 Edge Router	JunOS 24.2R1-S1.10	OSPF-MIB, JUNIPER-DOM-MIB, JUNIPER-MIB, BGP4-V2-MIB-JUNIPER, OSPFV3-MIB-JUNIPER	Certified
Mellanox SNMP	Mellanox	Urządzenia Mellanox Infiniband	SX1036	MLNX-OS	HOST-RESOURCES-MIB,ENTITY-MIB,ENTITY-SENSOR-MIB,MELLANOX-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System opera-cyjny	Używane MIB-y	**Tagi**
MikroTik CCR<device model> SNMP	MikroTik	Routery MikroTik Cloud Core (seria CCR)	Dostępne są oddzielne dedykowane szablony dla 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	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System opera-cyjny	Używane MIB-y	**Tagi**
MikroTik CRS<device model> SNMP	MikroTik	Przełączniki MikroTik Cloud Router (seria CRS)	Dostępne są oddzielne dedykowane szablony dla 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	Certified
MikroTik CSS<device model> SNMP	MikroTik	Przełączniki MikroTik Cloud Smart (seria CSS)	Dostępne są oddzielne dedykowane szablony dla MikroTik CSS326-24G-2S+RM, MikroTik CSS610-8G-2S+IN	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik FiberBox SNMP	MikroTik	MikroTik FiberBox	MikroTik FiberBox	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik hEX <device model> SNMP	MikroTik	MikroTik hEX	Dostępne są oddzielne dedykowane szablony dla MikroTik hEX, MikroTik hEX lite, MikroTik hEX PoE, MikroTik hEX PoE lite, MikroTik hEX S	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
MikroTik netPower <device model> SNMP	MikroTik	MikroTik netPower	Dostępne są oddzielne dedykowane szablony dla MikroTik netPower 15FR, MikroTik netPower 16P SNMP, MikroTik netPower Lite 7R	RouterOS/SwitchOS, SwitchOS Lite	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik PowerBox <device model> SNMP	MikroTik	MikroTik PowerBox	Dostępne są oddzielne dedykowane szablony dla MikroTik PowerBox, MikroTik PowerBox Pro	RouterOS	MIKROTIK-MIB,HOST-RESOURCES-MIB	Certified
MikroTik RB<device model> SNMP	MikroTik	Routery serii MikroTik RB	Dostępne są oddzielne dedykowane szablony dla 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	Certified
MikroTik SNMP	MikroTik	Urządzenia 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	Certified

Nazwa szablonu	Dostawca	Rodzina urządzeń	Znane modele	System operacyjny	Używane MIB-y	**Tagi**
Netgear Fastpath SNMP	Netgear	Netgear Fastpath	M5300-28G		FASTPATH-SWITCHING-MIB,FASTPATH-BOXSERVICES-PRIVATE-MIB	Fault Inventory
QTech QSW SNMP	QTech	Urządzenia Qtech	Qtech QSW-2800-28T	-	QTECH-MIB,ENTITY-MIB	Performance Inventory
Stormshield SNS by SNMP	Stormshield	Zapory 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	Certified
TP-LINK SNMP	TP-LINK	TP-LINK	T2600G-28TS v2.0		TPLINK-SYSMONITOR-MIB,TPLINK-SYSINFO-MIB	Performance Inventory
Ubiquiti AirOS SNMP	Ubiquiti	Bezprzewodowe urządzenia Ubiquiti AirOS	NanoBridge,NanoStation,Unifi	AirOS	FROGFOOT-RESOURCES-MIB,IEEE802dot11-MIB	Performance
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	Performance Inventory

Projekt szablonów

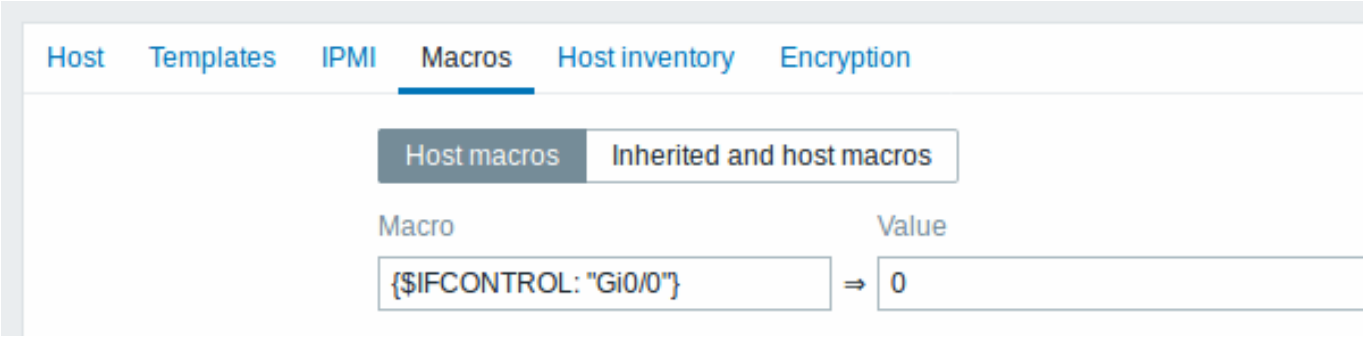
Szablony zostały zaprojektowane z uwzględnieniem następujących założeń:

- Makra użytkownika są używane w możliwie największym zakresie, aby wyzwalacze mogły być dostrajane przez użytkownika;
- Mechanizm wykrywania niskiego poziomu jest używany w możliwie największym zakresie, aby zminimalizować liczbę nieobsługiwanych pozycji;
- Wszystkie szablony zależą od Template ICMP Ping, dzięki czemu wszystkie urządzenia są również sprawdzane przez ICMP;
- Pozycje nie używają żadnych MIB-ów — w pozycjach i regułach wykrywania niskiego poziomu używane są OID-y SNMP. Dzięki temu nie ma potrzeby ładowania żadnych MIB-ów do Zabbix, aby szablony działały;
- Interfejsy sieciowe loopback są filtrowane podczas wykrywania, podobnie jak interfejsy z ifAdminStatus = down(2)
- Tam, gdzie to możliwe, używane są liczniki 64-bitowe z IF-MIB::ifXTable. Jeśli nie są obsługiwane, zamiast nich używane są domyślne liczniki 32-bitowe.

Wszystkie wykryte interfejsy sieciowe mają wyzwalacz monitorujący ich stan operacyjny (link), na przykład:

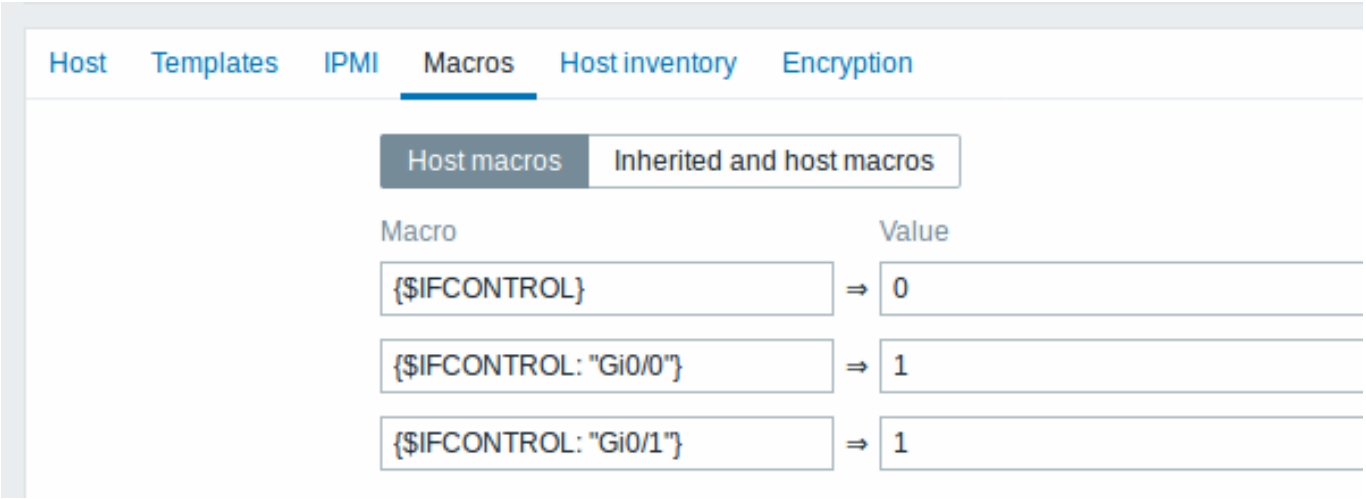
```
{$IFCONTROL:"{#IFNAME}"}=1 and last(/Alcatel Timetra TiMOS SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])
```

- Jeśli nie chcesz monitorować tego warunku dla określonego interfejsu, utwórz makro użytkownika z kontekstem o wartości 0. Na przykład:



gdzie Gi0/0 to {#IFNAME}. W ten sposób wyzwalacz nie będzie już używany dla tego konkretnego interfejsu.

- Możesz również zmienić domyślne zachowanie wszystkich wyzwalaczy tak, aby się nie uruchamiały, i aktywować ten wyzwalacz tylko dla ograniczonej liczby interfejsów, takich jak uplinki:



Tagi

- Wydajność - MIB-y rodziny urządzeń zapewniają sposób monitorowania pozycji procesora i pamięci;
- Błędy - MIB-y rodziny urządzeń zapewniają sposób monitorowania co najmniej jednego czujnika temperatury;
- Inwentaryzacja - MIB-y rodziny urządzeń zapewniają sposób zbierania co najmniej numeru seryjnego urządzenia i nazwy modelu;
- Certyfikowane - wszystkie trzy główne kategorie powyżej są objęte.

Operacja szablonu VMware

Przegląd

Zabbix zawiera szereg gotowych do użycia szablonów do monitorowania maszyn wirtualnych, przeznaczonych dla hiperwizorów VMware vCenter i ESXi. Dostępne szablony są zawarte w dwóch oddzielnych zestawach szablonów:

- VMware - używa danych UUID dla odpowiednich makr
- VMware FQDN - używa danych FQDN dla odpowiednich makr

Szablony przeznaczone do monitorowania hiperwizorów VMware vCenter lub ESXi zawierają wstępnie skonfigurowane reguły wykrywania niskiego poziomu wraz z różnymi wbudowanymi kontrolami do monitorowania instalacji wirtualnych.

Note:
Aby szablon *VMware FQDN* działał poprawnie, każda monitorowana maszyna wirtualna powinna mieć unikalną nazwę systemu operacyjnego zgodną z zasadami FQDN. Dodatkowo na każdej maszynie muszą być zainstalowane VMware Tools/Open Virtual Machine Tools. Jeśli te wymagania wstępne są spełnione, zalecane jest użycie szablonu *VMware FQDN*. Szablon *VMware FQDN* jest dostępny od Zabbix 5.2 wraz z wprowadzeniem możliwości tworzenia hostów z niestandardowymi interfejsami.

 Dostępny jest również klasyczny szablon *VMware*, którego można użyć, jeśli wymagania FQDN nie są spełnione. Jednak szablon *VMware* ma znany problem. Hosty dla wykrytych maszyn wirtualnych są tworzone z nazwami zapisanymi w vCenter (na przykład „VM1", „VM2" itd.). Jeśli na tych hostach zainstalowany jest agent Zabbix i włączona jest aktywna autorejestracja agent Zabbix, proces autorejestracji odczyta nazwy hostów tak, jak zostały zarejestrowane podczas uruchomienia (na przykład „vm1.example.com", „vm2.example.com" itd.). Może to prowadzić do tworzenia nowych hostów dla istniejących maszyn wirtualnych (ponieważ nie znaleziono zgodnych nazw), co skutkuje zduplikowanymi hostami o różnych nazwach.

Należy pamiętać:

- Szablon *VMware Hypervisor* może być ręcznie podłączony do hosta, a także używany w wykrywaniu;
- Szablon *VMware Guest* nie powinien być ręcznie podłączany do hosta i może być używany wyłącznie w wykrywaniu.

Konfiguracja makr hosta

Aby używać prostych kontroli VMware, host musi mieć zdefiniowane następujące makra użytkownika:

- {$VMWARE.URL} - adres URL SDK usługi VMware (vCenter lub hiperwizora ESXi) (https://servername/sdk)
- {$VMWARE.USERNAME} - nazwa użytkownika usługi VMware
- {$VMWARE.PASSWORD} - hasło użytkownika {$VMWARE.USERNAME} usługi VMware

Działanie szablonów OTLP

Zabbix udostępnia szereg gotowych szablonów do zbierania i monitorowania śladów, metryk oraz dzienników OpenTelemetry za pomocą pozycji typu *Telemetry query*.

Kroki zapewniające prawidłowe działanie szablonów:

1. Sprawdź plik `README` szablonu pod kątem wymagań dotyczących konfiguracji właściwych dla danego szablonu.
2. Utwórz host w Zabbix odpowiadający monitorowanemu celowi.
3. Upewnij się, że parametr `TelemetryProvider` jest skonfigurowany w pliku konfiguracyjnym Zabbix serwera lub Zabbix proxy.
4. Połącz szablon z utworzonym hostem (jeśli szablon nie jest dostępny w używanej instalacji Zabbix, może być konieczne jego wcześniejsze zaimportowanie).
5. W razie potrzeby dostosuj wartości makr na poziomie szablonu.

Szczegółowy opis szablonów, w tym pełna lista makr, pozycji i wyzwalaczy, jest dostępny w pliku `README` szablonu. Możesz go otworzyć, klikając nazwę szablonu.

Dostępne są następujące szablony:

- Envoy Proxy by OTLP
- Generic OpenTelemetry by OTLP
- OpenRouter traces by OTLP
- Proxmox VE by OTLP

Powiadomienia po wystąpieniu zdarzeń

Przegląd

Zakładając, że skonfigurowaliśmy już pewne pozycje i wyzwalacze, a teraz otrzymujemy zdarzenia występujące w wyniku zmiany stanu wyzwalaczy, nadszedł czas, aby rozważyć działania.

Na początek nie chcielibyśmy cały czas wpatrywać się w listę wyzwalaczy lub zdarzeń. Znacznie lepiej byłoby otrzymywać powiadomienie, jeśli wydarzyło się coś istotnego (na przykład problem). Ponadto, gdy wystąpią problemy, chcielibyśmy mieć pewność, że wszystkie zainteresowane osoby zostały poinformowane.

Dlatego wysyłanie powiadomień jest jednym z podstawowych działań oferowanych przez Zabbix. Można określić, kto i kiedy powinien zostać powiadomiony o określonym zdarzeniu.

Aby móc wysyłać i odbierać powiadomienia z Zabbix, należy:

- zdefiniować media
- skonfigurować działanie, które wysyła wiadomość do jednego ze zdefiniowanych mediów

Działania składają się z *warunków* i *operacji*. Zasadniczo, gdy warunki są spełnione, wykonywane są operacje. Dwie główne operacje to wysłanie wiadomości (powiadomienia) oraz wykonanie zdalnego polecenia.

W przypadku zdarzeń utworzonych przez wykrywanie i autorejestrację dostępne są dodatkowe operacje. Obejmują one dodawanie lub usuwanie hosta, podłączanie szablonu itp.

Typy mediów

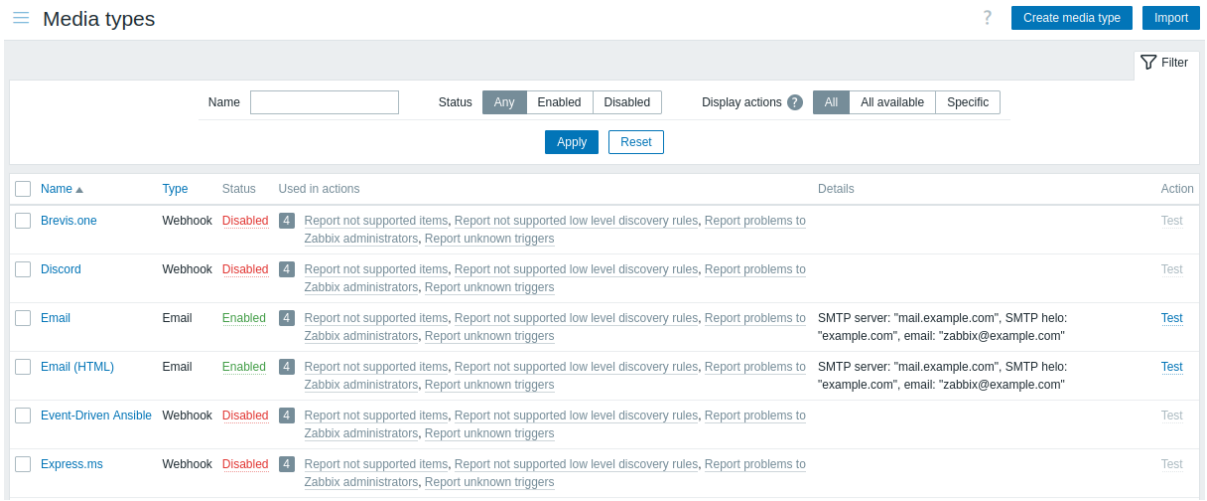
Przegląd

Typy mediów to kanały dostarczania używane do wysyłania powiadomień i alertów z systemu Zabbix.

Typy mediów obsługują następujące metody dostarczania:

- E-mail
- Powiadomienie push
- SMS
- Skrypt niestandardowy
- Webhook

Typy mediów są zarządzane w sekcji *Alerty* > *Typy mediów*. Niektóre typy mediów są wstępnie zdefiniowane w domyślnym zestawie danych. Wystarczy dostosować ich parametry, aby zaczęły działać.



Aby zobaczyć, jak typy mediów wpisują się w proces alertowania, przyjrzyjmy się trzem wymaganiom dotyczącym dostarczania powiadomień z systemu Zabbix do użytkowników końcowych:

1. Należy zdefiniować operację akcji, która wysyła powiadomienia.
2. Należy zdefiniować działający typ mediów, taki jak *E-mail*, który wysyła alerty za pomocą SMTP.
3. Należy zdefiniować dane dotyczące dostarczania na poziomie użytkownika, takie jak adresy e-mail, numery telefonów itp., w sekcji media użytkownika.

Testowanie typu nośnika

Aby sprawdzić, czy skonfigurowany typ nośnika działa, kliknij *Test* na liście typów nośników.

Żądanie testowe zostanie wysłane do serwera Zabbix. Serwer Zabbix spróbuje wysłać alert przy użyciu określonego typu nośnika i zwróci wynik do frontend. Frontend będzie oczekiwać na zwrócenie wyników przez serwer. Testowanie typu nośnika ma domyślnie limit czasu wynoszący 65 sekund (można go skonfigurować w sekcji *Administracja* > *Ogólne* > *Limity czasu*).

Zobacz także szczegóły testowania dla:

- Email
- Powiadomienie push
- Webhook
- Skrypt

Konfiguracja

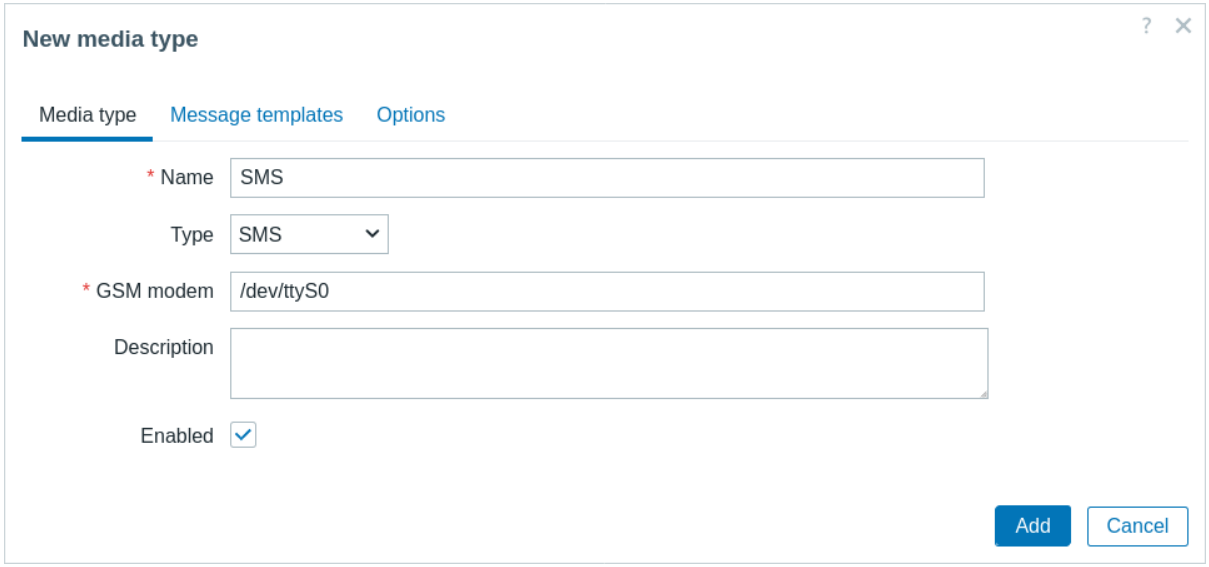
Aby utworzyć typ nośnika w frontend Zabbix:

1. Przejdź do *Alerts* > *Media types*.

2. Kliknij *Create media type*.
3. Wprowadź parametry typu nośnika w formularzu.

Niektóre parametry są **wspólne** dla wszystkich metod dostarczania.

Note:
Jeśli co najmniej jedna metoda dostarczania została określona we fladze funkcji `$ZBX_FEATURE_FLAGS['media_type_denylist']` w pliku konfiguracyjnym frontendu (`zabbix.conf.php`), te metody dostarczania nie są dostępne na liście *Type*.

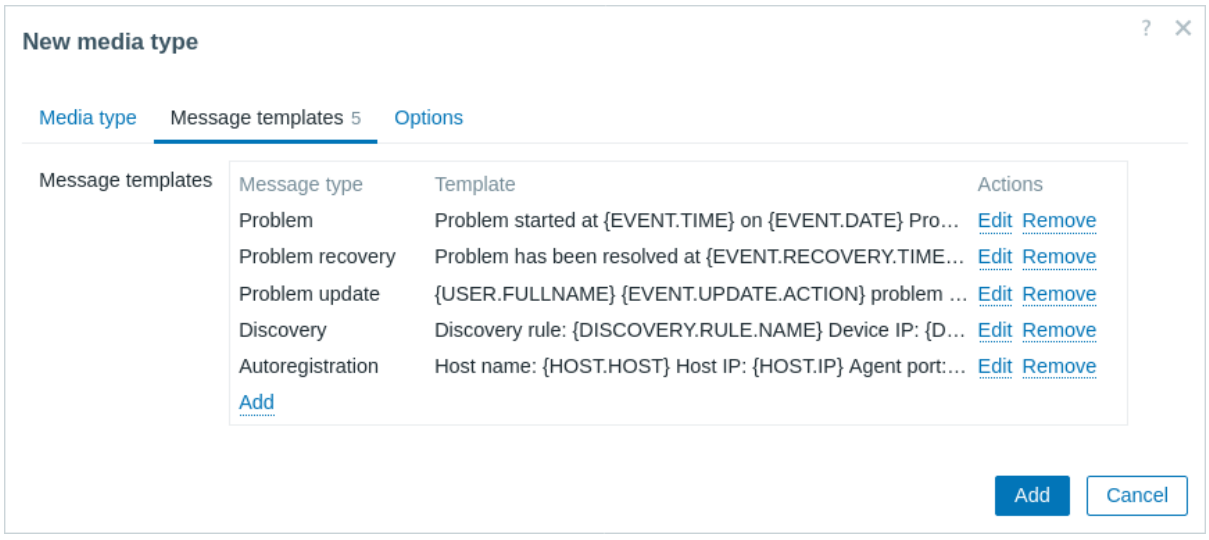


Parameter	Description
Name	Nazwa typu nośnika.
Type	Wybierz metodę dostarczania dla typu nośnika.
Description	Wprowadź opis typu nośnika.
Enabled	Zaznacz pole wyboru, aby włączyć typ nośnika.

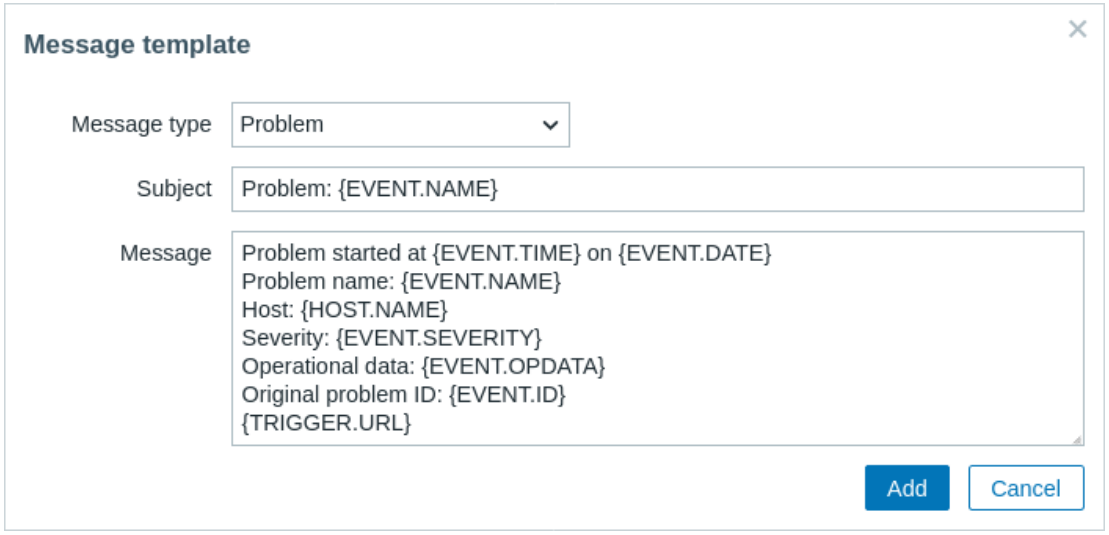
Parametry specyficzne dla poszczególnych metod opisano na stronach dotyczących poczty e-mail, powiadomień push, SMS, niestandardowego alertscriptu lub webhooka.

Szablony wiadomości

Karta **Message templates** zawiera domyślne wiadomości dla poszczególnych typów zdarzeń (problem, odzyskanie po problemie, wykrywanie itp.).



Kliknij **Add**, aby zdefiniować domyślną wiadomość (lub **Edit**, aby zaktualizować istniejącą wiadomość):



Parameter	Description
Message type	Typ zdarzenia, dla którego ma być używana domyślna wiadomość. Dla każdego typu zdarzenia można zdefiniować tylko jedną domyślną wiadomość.
Subject	Temat domyślnej wiadomości. Temat może zawierać makra. Jego długość jest ograniczona do 255 znaków. Temat nie jest dostępny dla typu nośnika SMS.
Message	Domyślna wiadomość. Jej długość jest ograniczona do określonej liczby znaków w zależności od typu bazy danych (więcej informacji można znaleźć w sekcji Sending messages). Wiadomość może zawierać obsługiwane makra. W wiadomościach o problemie i aktualizacji problemu obsługiwane są makra wyrażeń (na przykład `{?avg(/host/key,1h)}`).

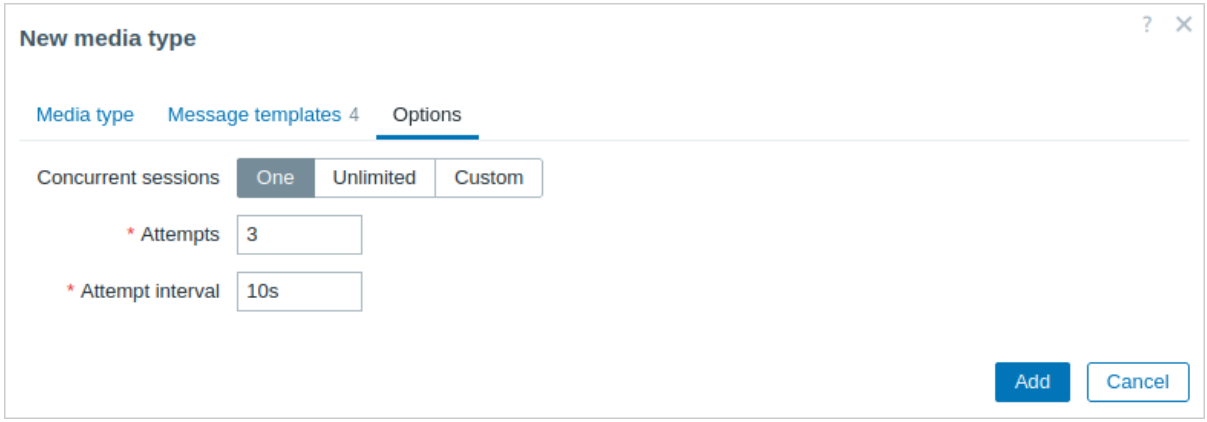
Należy pamiętać, że domyślne wiadomości są zastępowane przez wiadomości niestandardowe, jeśli zostały zdefiniowane w action operations.

Warning:
Definiowanie szablonów wiadomości jest obowiązkowe dla wszystkich metod dostarczania, w tym webhooków lub niestandardowych skryptów alertów, które nie używają domyślnych wiadomości do powiadomień. Na przykład akcja "Send message to Pushover webhook" nie wyśle powiadomień o problemie, jeśli wiadomość o problemie dla Pushover webhook nie została zdefiniowana.

Opcje

Karta **Opcje** zawiera ustawienia przetwarzania alertów.
Ten sam zestaw opcji można skonfigurować dla każdego typu nośnika.

Wszystkie typy nośników są przetwarzane równolegle.
Maksymalną liczbę jednoczesnych sesji można skonfigurować dla każdego typu nośnika,
natomiast całkowitą liczbę procesów alertera na serwerze można ograniczyć wyłącznie za pomocą parametru StartAlerters.
Alerty wygenerowane przez jeden wyzwalacz są przetwarzane sekwencyjnie.
Oznacza to, że wiele powiadomień może być przetwarzanych jednocześnie tylko wtedy, gdy zostały wygenerowane przez wiele wyzwalaczy.



Parametr	Opis
Jednoczesne sesje	Wybierz liczbę równoległych sesji alertera dla typu nośnika: **Jedna** - jedna sesja **Bez ograniczeń** - nieograniczona liczba sesji **Niestandardowa** - wybierz niestandardową liczbę sesji Brak ograniczeń/duże wartości oznaczają większą liczbę równoległych sesji i większą wydajność wysyłania powiadomień. Brak ograniczeń/duże wartości należy stosować w dużych środowiskach, w których może być konieczne jednoczesne wysłanie wielu powiadomień. Jeśli liczba powiadomień do wysłania przekracza liczbę jednoczesnych sesji, pozostałe powiadomienia zostaną umieszczone w kolejce; nie zostaną utracone.
Próby	Liczba prób wysłania powiadomienia. Można określić maksymalnie 100 prób; wartość domyślna to „3”. Jeśli określono „1”, Zabbix wyśle powiadomienie tylko raz i nie ponowi próby w przypadku niepowodzenia wysyłania.
Interwał między próbami	Częstotliwość ponawiania próby wysłania powiadomienia w przypadku niepowodzenia wysyłania, w sekundach (0-3600). Jeśli określono „0”, Zabbix ponowi próbę natychmiast. Obsługiwane są przyrostki czasu (np. 30s, 1m, 1h).

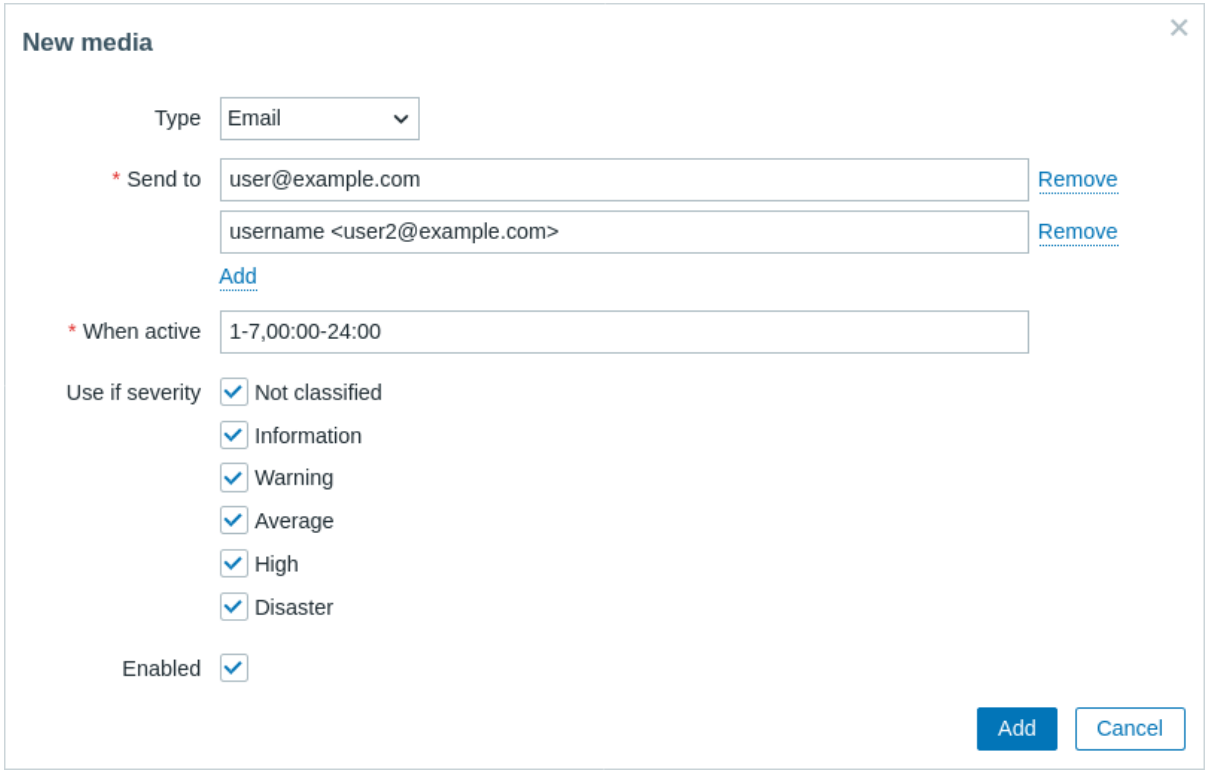
Media użytkownika

Podczas gdy typy mediów określają **sposób** wysyłania powiadomienia, media użytkownika określają **miejsce**, do którego powiadomienie musi zostać wysłane.

Media użytkownika (np. adres e-mail, identyfikator użytkownika webhooka itp.) muszą być zdefiniowane w profilu użytkownika niezależnie od metody dostarczania. Akcja wysyłająca wiadomości do użytkownika *Admin* za pomocą webhooka *X* nie dostarczy wiadomości, jeśli szczegóły dostarczania webhooka *X* nie są zdefiniowane w profilu użytkownika Admin.

Aby zdefiniować media użytkownika:

1. Przejdź do *Użytkownicy* > *Użytkownicy* i otwórz formularz właściwości użytkownika. Możesz też przejść do sekcji *Ustawienia użytkownika* > *Powiadomienia* we własnym profilu użytkownika.
2. Kliknij *Dodaj* na karcie *Media*.
3. Wprowadź parametry mediów użytkownika w formularzu.



Parametr	Opis
Typ	Lista rozwijana zawiera nazwy włączonych typów mediów. Pamiętaj, że podczas edycji medium wyłączonego typu mediów typ zostanie wyświetlony na czerwono.
Wyślij do	Wprowadź dane kontaktowe, na które powinny być wysyłane wiadomości. W przypadku typu mediów e-mail można dodać wiele adresów, klikając przycisk Add poniżej pola adresu. W takim przypadku powiadomienia będą wysyłane na wszystkie wymienione adresy. Przykłady adresów znajdziesz w opisie parametru *E-mail* dla typu mediów email. W przypadku typu mediów powiadomień push możesz wybrać aktywne urządzenia mobilne użytkownika lub dodać co najmniej jedno konkretne urządzenie.
Aktywne w okresie	Możesz ograniczyć czas wysyłania wiadomości, na przykład ustawić tylko dni robocze (1-5,09:00-18:00). Pamiętaj, że ograniczenie to jest oparte na strefie czasowej użytkownika. Jeśli strefa czasowa użytkownika zostanie zmieniona i będzie różnić się od strefy czasowej systemu, ograniczenie może wymagać odpowiedniego dostosowania, aby nie pominąć ważnych wiadomości. Opis formatu znajdziesz na stronie Specyfikacja okresu. Obsługiwane są makra użytkownika.
Użyj, jeśli ważność	Zaznacz pola wyboru ważności wyzwalaczy, dla których chcesz otrzymywać powiadomienia. *Uwaga*: domyślna ważność („Niesklasyfikowane”) **musi** być zaznaczona, jeśli chcesz otrzymywać powiadomienia dotyczące zdarzeń innych niż wyzwalacze. Po zapisaniu wybrane ważności wyzwalaczy będą wyświetlane w odpowiadających im kolorach ważności, a niezaznaczone zostaną wyszarzone.
Status	Status mediów użytkownika. **Włączone** - jest używane. **Wyłączone** - nie jest używane.

E-mail

Przegląd

Aby skonfigurować e-mail jako kanał dostarczania wiadomości, należy skonfigurować e-mail jako typ mediów i przypisać użytkownikom określone adresy.

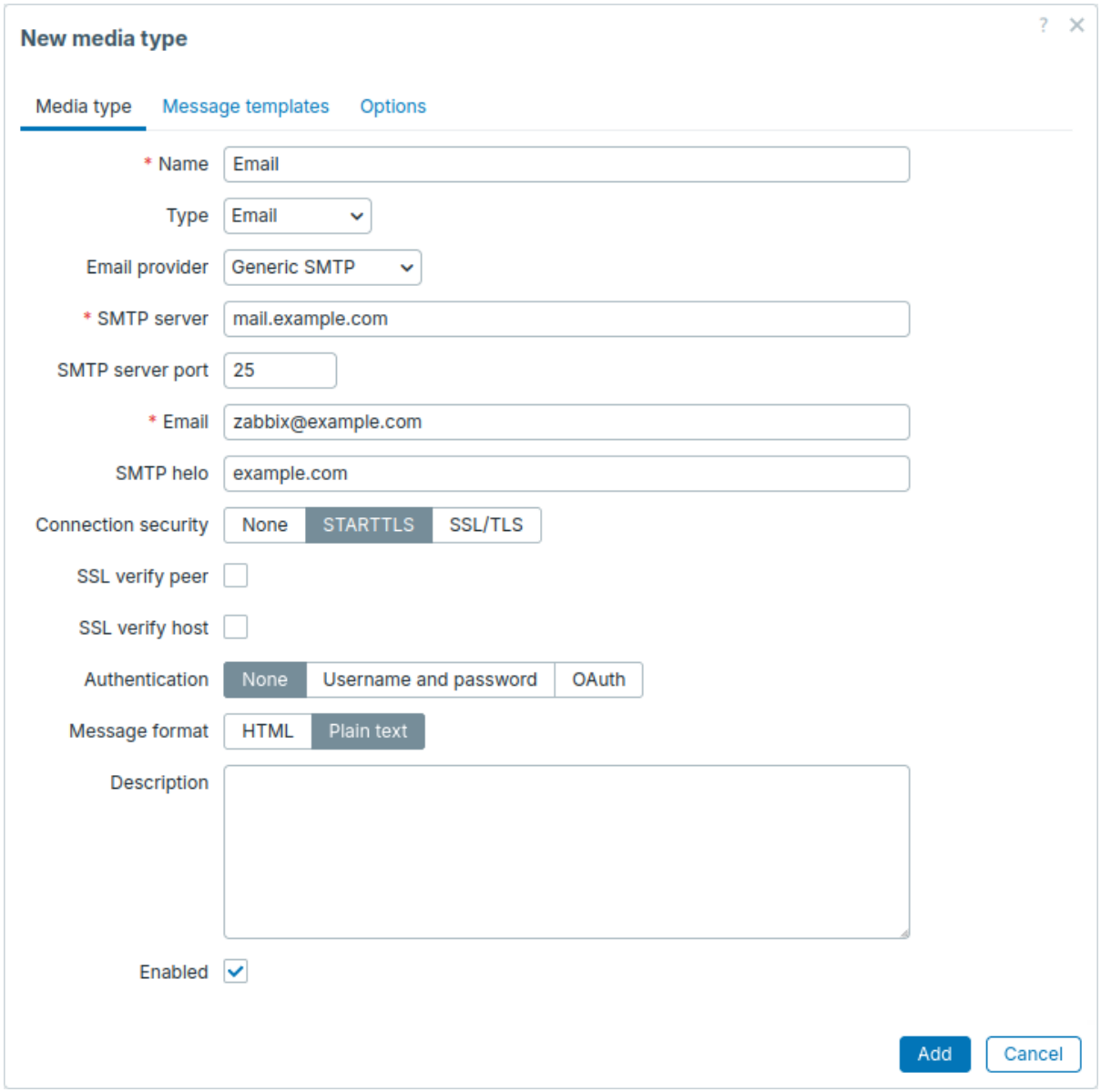
Note:
Wiele powiadomień dla pojedynczego zdarzenia zostanie zgrupowanych w tym samym wątku e-mail.

Konfiguracja

Aby skonfigurować pocztę e-mail jako typ nośnika:

1. Przejdź do *Alerts* > *Media types*.
2. Kliknij *Create media type* (lub kliknij *Email* na liście wstępnie zdefiniowanych typów nośników).
3. Wprowadź parametry typu nośnika e-mail w formularzu.

Karta **Media type** zawiera ogólne atrybuty typu nośnika:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Warning:
Hasło jest eksportowane w postaci jawnego tekstu podczas eksportowania konfiguracji typu nośnika e-mail.

Poniższe parametry są specyficzne dla typu nośnika e-mail:

Parameter	Description
Email provider	Wybierz dostawcę poczty e-mail: *Generic SMTP*, *Gmail*, *Gmail relay*, *Office365* lub *Office365 relay*. Jeśli wybierzesz opcję związaną z Gmail/Office365, wystarczy podać adres e-mail nadawcy i hasło; opcje takie jak *SMTP server*, *SMTP server port*, *SMTP helo* i *Connection security* zostaną automatycznie wypełnione przez Zabbix. Zobacz także: Automated Gmail/Office365 media types.
SMTP server	Ustaw serwer SMTP do obsługi wiadomości wychodzących. To pole jest dostępne, jeśli jako dostawcę poczty e-mail wybrano *Generic SMTP*.

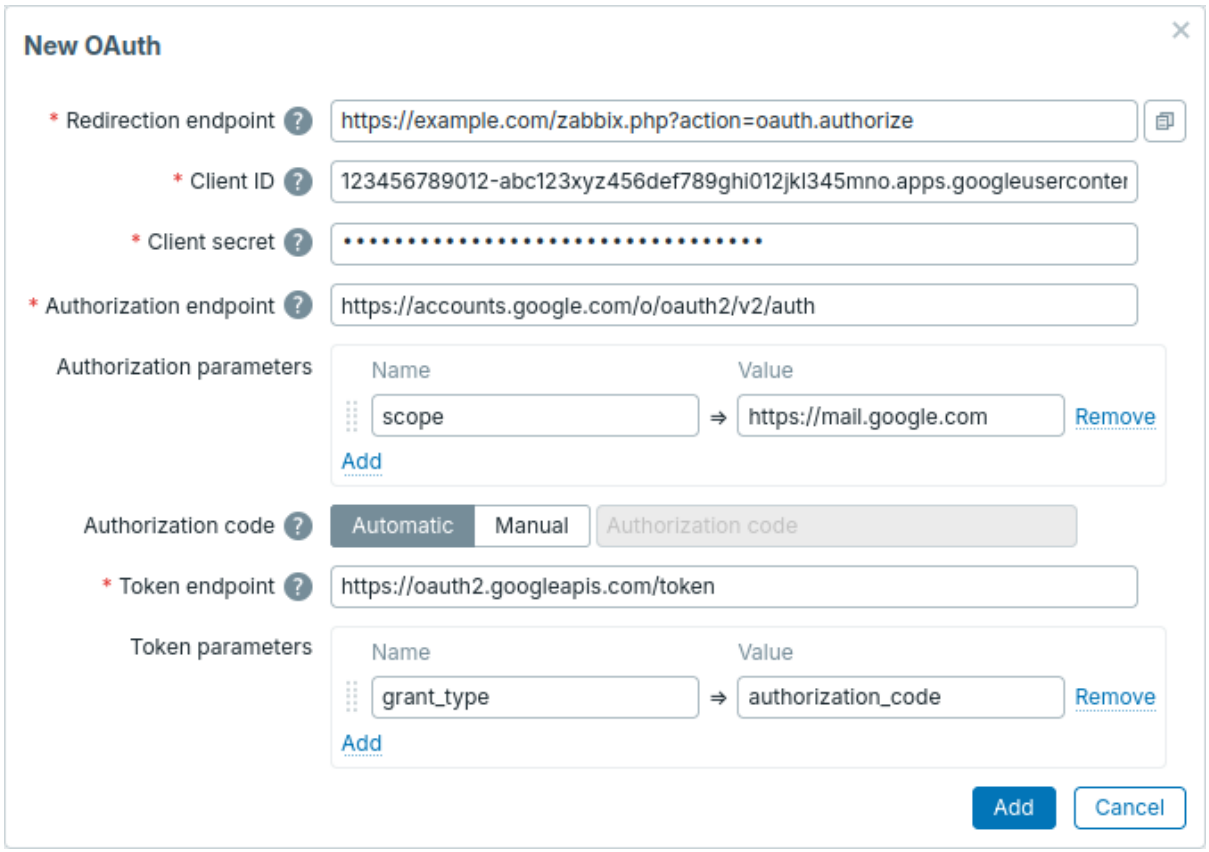
Parameter	Description
SMTP server port	Ustaw port serwera SMTP do obsługi wiadomości wychodzących. To pole jest dostępne, jeśli jako dostawcę poczty e-mail wybrano *Generic SMTP*.
Email	Wprowadzony tutaj adres będzie używany jako adres **From** dla wysyłanych wiadomości. Obsługiwane jest dodanie wyświetlanej nazwy nadawcy wraz z rzeczywistym adresem e-mail (na przykład `Zabbix_info <zabbix@company.com>`). Istnieją pewne ograniczenia dotyczące wyświetlanych nazw w wiadomościach e-mail Zabbix w porównaniu z nazwami dozwolonymi przez RFC 5322, jak pokazano w poniższych przykładach. Przykłady prawidłowe: `zabbix@company.com` (tylko adres e-mail; nawiasy ostrokątne nie są wymagane) `Zabbix_info <zabbix@company.com>` (wyświetlana nazwa, po której następuje adres e-mail w nawiasach ostrokątnych) `Ω-monitoring <zabbix@company.com>` (znaki UTF-8 w wyświetlanej nazwie) Przykłady nieprawidłowe: `Zabbix HQ zabbix@company.com` (wyświetlana nazwa jest obecna, ale wokół adresu e-mail nie ma nawiasów ostrokątnych) `"Zabbix@<H(comment)Q>" <zabbix@company.com>` (mimo że jest to prawidłowe zgodnie z RFC 5322, cytowane pary i komentarze nie są obsługiwane w wiadomościach e-mail Zabbix).
SMTP helo	Ustaw prawidłową wartość SMTP helo, zwykle nazwę domeny. Jeśli pole pozostanie puste, zostanie wysłana nazwa domeny adresu e-mail (czyli część znajdująca się po `@` w polu *Email*). Jeśli pobranie nazwy domeny będzie niemożliwe, zostanie zarejestrowane ostrzeżenie na poziomie debugowania, a nazwa hosta serwera zostanie wysłana jako domena dla polecenia HELO. To pole jest dostępne, jeśli jako dostawcę poczty e-mail wybrano *Generic SMTP*.
Connection security	Wybierz poziom bezpieczeństwa połączenia: **None** - nie używaj opcji CURLOPT_USE_SSL **STARTTLS** - użyj opcji CURLOPT_USE_SSL z wartością CURLUSESSL_ALL **SSL/TLS** - użycie CURLOPT_USE_SSL jest opcjonalne.
SSL verify peer	Zaznacz pole wyboru, aby zweryfikować certyfikat SSL serwera SMTP. Wartość dyrektywy konfiguracji serwera ”SSLCALocation” należy umieścić w CURLOPT_CAPATH w celu walidacji certyfikatu. Ustawia to opcję cURL CURLOPT_SSL_VERIFYPEER.
SSL verify host	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub *Subject Alternate Name* certyfikatu serwera SMTP jest zgodne. Ustawia to opcję cURL CURLOPT_SSL_VERIFYHOST.
Authentication	Wybierz poziom uwierzytelniania: **None** - opcje cURL nie są ustawiane **Username and password** - oznacza ”AUTH=*”, pozostawiając cURL wybór mechanizmu uwierzytelniania **OAuth** - uwierzytelnianie OAuth Uwierzytelnianie OAuth nie jest obsługiwane przez dostawcę poczty e-mail *Office365 relay*.
Username	Nazwa użytkownika używana podczas uwierzytelniania. Ustawia wartość CURLOPT_USERNAME. Obsługiwane są makra użytkownika.
Password	Hasło używane podczas uwierzytelniania. Ustawia wartość CURLOPT_PASSWORD. Obsługiwane są makra użytkownika. Wprowadzona tutaj wartość jest eksportowana bez zmian podczas eksportowania konfiguracji typu nośnika. Aby uniknąć ujawnienia poufnych danych uwierzytelniających w eksportowanych plikach, użyj makra użytkownika (na przykład `{$EMAIL_SMTP_PASSWORD}`) zamiast hasła wpisanego bezpośrednio — podczas eksportowania zostanie uwzględnione odwołanie do makra, natomiast wartość sekretu powinna zostać skonfigurowana lub skonfigurowana ponownie w systemie docelowym.
OAuth tokens	Kliknij *Configure*, aby skonfigurować parametry pobierania tokenów OAuth w nowym oknie. To pole jest dostępne tylko wtedy, gdy w polu *Authentication* wybrano opcję ”OAuth”.
Message format	Wybierz format wiadomości: **HTML** - wysyłaj jako HTML; **Plain text** - wysyłaj jako zwykły tekst.

Attention:
Aby włączyć uwierzytelnianie SMTP, serwer Zabbix musi zostać skompilowany z opcją `--with-libcurl` kompilacji, która wymaga biblioteki libcurl (szczegółowe informacje o wersjach znajdują się w wymaganiach dotyczących serwera).

Zobacz także common media type parameters, aby uzyskać szczegółowe informacje na temat konfigurowania domyślnych wiadomości i opcji przetwarzania alertów.

Tokeny OAuth

Do pobrania tokenów OAuth wymagane są następujące parametry:



Warning:
Client secret jest eksportowany w postaci jawnego tekstu podczas eksportowania konfiguracji typu mediów e-mail.

Parameter	Description
Redirection endpoint	Wprowadź URL frontend Zabbix, na który usługa OAuth przekieruje odpowiedź autoryzacji OAuth (w formacie `https://<zabbix-frontend-url>/zabbix.php?action=oauth.authorize`). Jest on ustawiany automatycznie dla nowego typu mediów e-mail, gdy zdefiniowany jest frontend URL.
Client ID	Wprowadź unikalny identyfikator aplikacji klienckiej zarejestrowanej w serwerze autoryzacji OAuth.
Client secret	Wprowadź prywatny sekret aplikacji klienckiej zarejestrowanej w serwerze autoryzacji OAuth.
Authorization endpoint	Wprowadź URL serwera autoryzacji OAuth, aby zażądać autoryzacji użytkownika.
Authorization parameters	Wprowadź parametry dla punktu końcowego autoryzacji.
Authorization code	Wprowadź kod autoryzacyjny: **Automatic** - kod zostanie pobrany automatycznie za pośrednictwem strony przekierowania; **Manual** - wprowadź kod ręcznie, jeśli automatyczne pobieranie się nie powiedzie.
Token endpoint	Wprowadź URL serwera autoryzacji OAuth, aby wymienić kod autoryzacyjny na tokeny dostępu i odświeżania.
Token parameters	Wprowadź parametry dla tokenu dostępu.

Note:
Pobieranie tokenów OAuth ma zautomatyzowane funkcje dla dostawców poczty **Gmail**, **Gmail relay** i **Office365**. Wymagane jest jedynie podanie wartości parametrów *Redirection endpoint*, *Client ID* i *Client secret*. Zabbix automatycznie uzupełni pozostałe wymagane wartości (zobacz Domyślne adresy URL OAuth według dostawcy).

Pobieranie tokenów dostępu i odświeżania OAuth

Po przesłaniu parametrów OAuth:

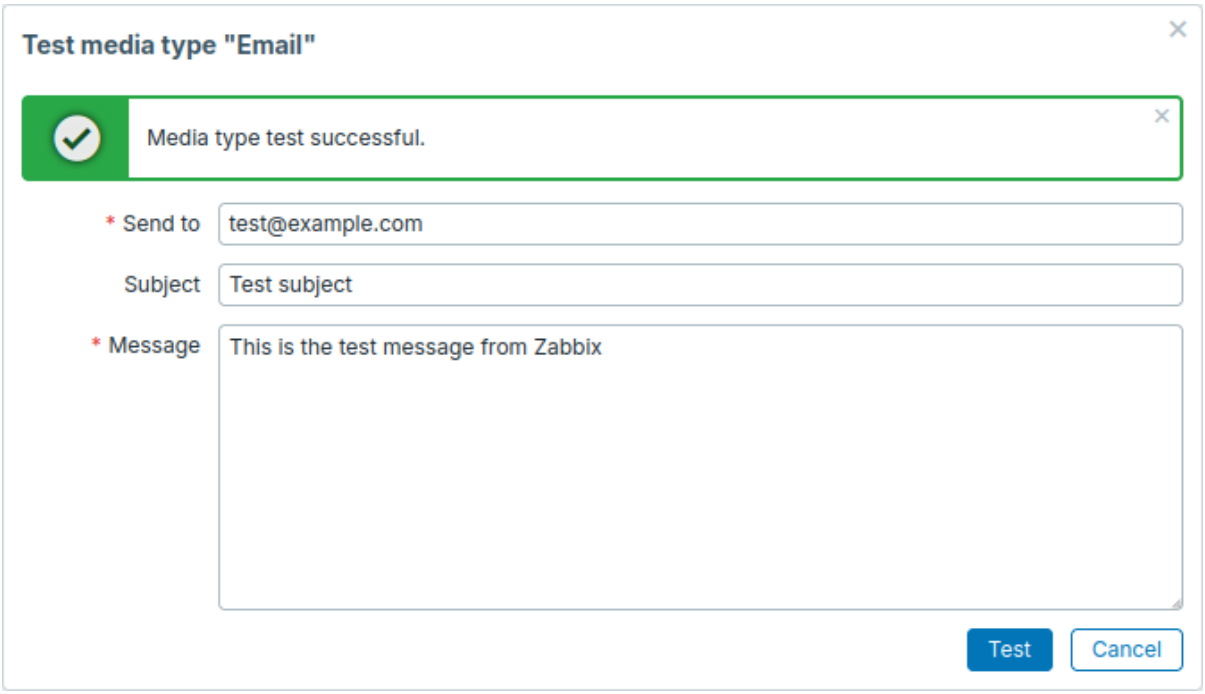
1. Otwierane jest wyskakujące okno przeglądarki, które kieruje użytkownika do *punktu końcowego autoryzacji*.
2. Użytkownik autoryzuje Zabbix w usłudze OAuth.
3. Usługa OAuth przekierowuje użytkownika do akcji frontendu Zabbix `oauth.authorize` z kodem autoryzacyjnym i wartością zakresu.
4. W odpowiedzi Zabbix wyśle żądanie do *punktu końcowego tokenu*, aby wymienić kod autoryzacyjny na token dostępu i token odświeżania.

Testowanie

Aby sprawdzić, czy skonfigurowany typ mediów e-mail działa poprawnie:

1. Znajdź odpowiedni e-mail na liście typów mediów.
2. Kliknij *Test* w ostatniej kolumnie listy (otworzy się okno testowe).
3. Wprowadź adres odbiorcy w polu *Send to*, treść wiadomości oraz opcjonalnie temat.
4. Kliknij *Test*, aby wysłać wiadomość testową.

W tym samym oknie zostanie wyświetlony komunikat o powodzeniu lub niepowodzeniu testu:



Media użytkownika

Po skonfigurowaniu typu mediów e-mail przejdź do sekcji *Users* > *Users* i edytuj profil użytkownika, aby przypisać mu media e-mail. Kroki konfiguracji mediów użytkownika, wspólne dla wszystkich typów mediów, są opisane na stronie Media types.

Automatyczne typy mediów Gmail/Office365

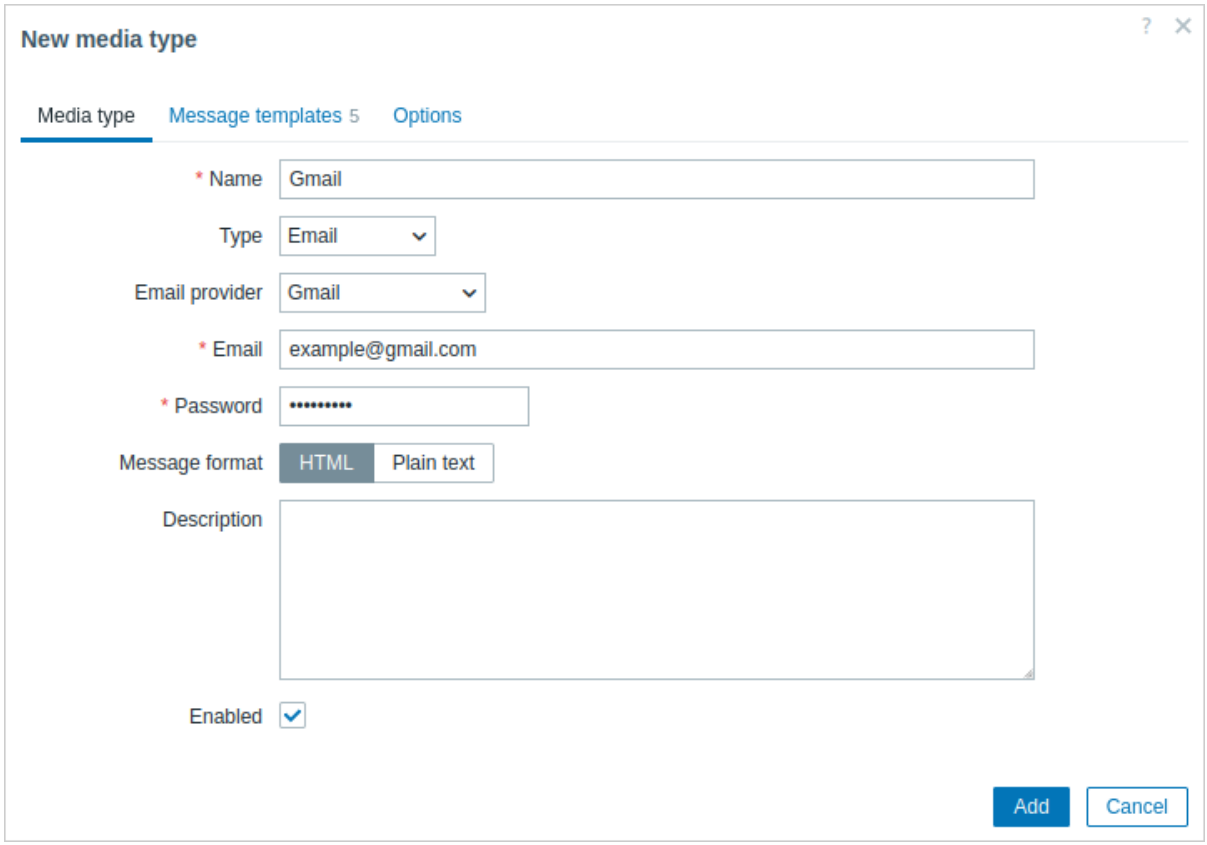
Przegląd

Użytkownicy Gmaila lub Office365 mogą skorzystać z automatycznych funkcji w konfiguracji typu mediów.

Konfiguracja

Pole *Dostawca poczty* w konfiguracji typu nośnika wiadomości e-mail umożliwia wybranie wstępnie skonfigurowanych opcji dla Gmaila i Office 365.

W przypadku wybrania opcji związanych z Gmailem lub Office 365 wystarczy podać adres e-mail nadawcy i hasło, aby utworzyć działający typ nośnika.



Po podaniu adresu e-mail i hasła Zabbix będzie mógł automatycznie uzupełnić wszystkie wymagane ustawienia typów nośników Gmaila i Office 365 rzeczywistymi lub zalecanymi wartościami, tj. *serwer SMTP*, *port serwera SMTP*, *helo SMTP* oraz *Zabezpieczenia połączenia*. Ze względu na tę automatyzację pola te nie są nawet wyświetlane, jednak szczegóły serwera SMTP i wiadomości e-mail można zobaczyć na liście typów nośników (w kolumnie *Szczegóły*).

Należy również pamiętać, że:

- Hasło nie jest wymagane w przypadku opcji przekazywania.
- W przypadku przekazywania za pośrednictwem Office 365 nazwa domeny podanego adresu e-mail zostanie użyta do dynamicznego uzupełnienia serwera SMTP (tj. zastąpienia ciągu „example.com" w `example-com.mail.protection.outlook.com` rzeczywistą wartością).

Tokeny OAuth

Pobieranie tokenów OAuth ma zautomatyzowane funkcje dla dostawców poczty e-mail *Gmail*, *Gmail relay* i *Office365*.



Konieczne jest podanie jedynie ograniczonego zestawu parametrów do uzyskania dostępu OAuth oraz pobrania tokenu odświeżania — *Redirection endpoint*, *Client ID* i *Client secret*. Zabbix automatycznie uzupełni pozostałe wymagane wartości (zobacz Domyślne adresy URL OAuth według dostawcy).

Należy pamiętać, że możliwe jest również użycie formularza Generic SMTP do skonfigurowania autoryzacji OAuth dla tych dostawców.

Note:
SmtpClientAuthentication musi być włączone albo dla użytkownika/współdzielonej skrzynki pocztowej, albo dla całej dzierżawy w Office365 (domyślnie nie jest włączone).

Domyślne adresy URL OAuth według dostawcy

Poniższa tabela zawiera domyślne wartości adresów URL OAuth i parametrów dla poszczególnych dostawców używane przez Zabbix.

Parametr	Gmail	Office365	Generic SMTP
Punkt końcowy autoryzacji	`https://accounts.google.com/o/oauth2/auth`	`https://login.microsoftonline.com/common/oauth2/v2.0/authori`	Brak wartości domyślnej
scope	`https://mail.google.com`	`https://outlook.office.com/SMTP.Send offline_access`	Brak wartości domyślnej
access_type	`offline`	Nieużywane	Brak wartości domyślnej
prompt	`consent`	Nieużywane	Brak wartości domyślnej
redirect_uri	Wartość pola *Redirection endpoint* jest dodawana automatycznie przez kod. Nie jest wymieniona w parametrach *Authorization endpoint*.		
state	Unikalny hash identyfikujący nośnik aktualizowany tokenem OAuth jest dodawany automatycznie przez kod. Nie jest wymieniony w parametrach *Authorization endpoint*.		
Punkt końcowy tokenu	`https://oauth2.googleapis.com/token`	`https://login.microsoftonline.com/common/oauth2/v2.0/token`	Brak wartości domyślnej
grant_type	`authorization_code`	`authorization_code`	Brak wartości domyślnej
redirect_uri	Wartość pola *Redirection endpoint* jest dodawana automatycznie przez kod. Nie jest wymieniona w parametrach *Token endpoint*.		
state	Unikalny hash identyfikujący nośnik aktualizowany tokenem OAuth jest dodawany automatycznie przez kod. Nie jest wymieniony w parametrach *Token endpoint*.		

Obejście dla kont osobistych Office365

Konta osobiste Office365 nie obsługują ciągów zapytań w adresie URL przekierowania.

Aby obejść ten problem, administrator serwera WWW Apache może dodać następującą regułę przepisywania do pliku `.htaccess`:

```
RewriteEngine On
RewriteRule ^/oauth\.authorize$ /zabbix.php?action=oauth.authorize [QSA,L,PT]
```

Umożliwia to użycie uproszczonego adresu URL przekierowania, takiego jak:

```
http://server-name/zabbix/oauth.authorize
```

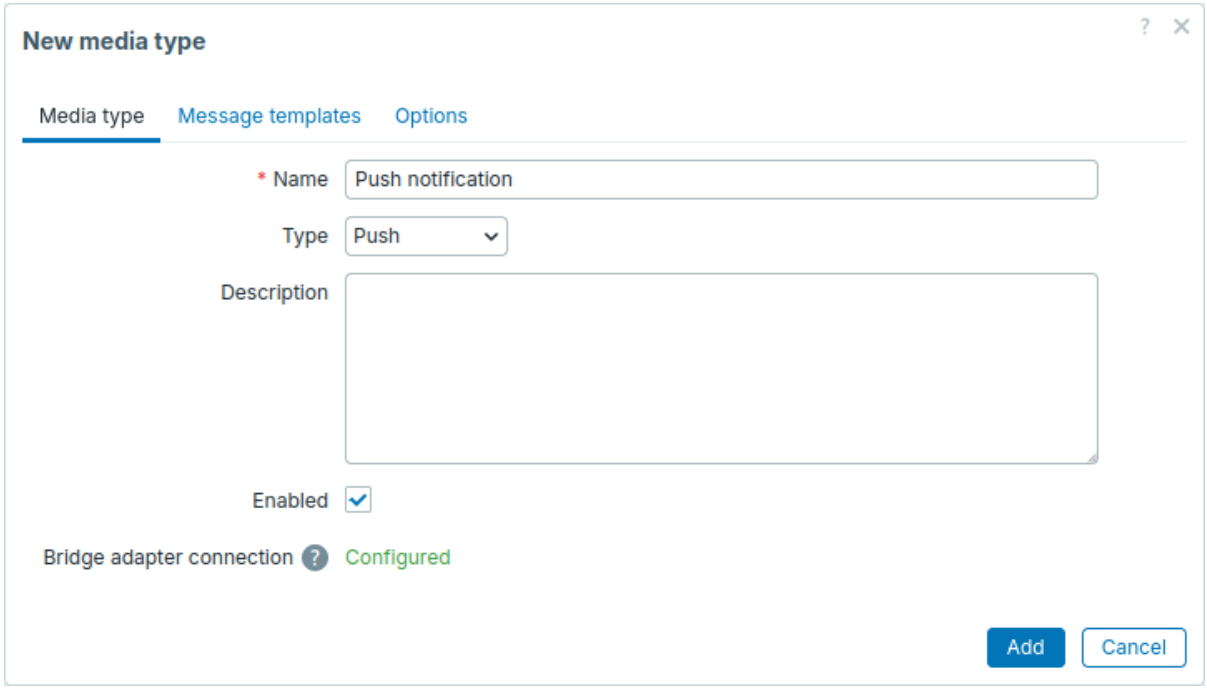
zamiast adresu URL z parametrami zapytania, co zapewnia zgodność z kontami osobistymi Office365.

Powiadomienia push

Ten typ nośnika umożliwia wysyłanie przez serwer Zabbix powiadomień push do aplikacji mobilnej Zabbix na urządzeniu mobilnym.

Konfiguracja

Podczas tworzenia typu nośnika wybierz *Push* jako jego *Typ*:



Formularz konfiguracji zawiera wspólne parametry typu nośnika, szablony wiadomości oraz opcje.

Formularz pokazuje również, czy adapter Zabbix bridge jest skonfigurowany.
 Jeśli adapter nie jest skonfigurowany, nadal możesz utworzyć typ nośnika, ale nie będzie on działać.

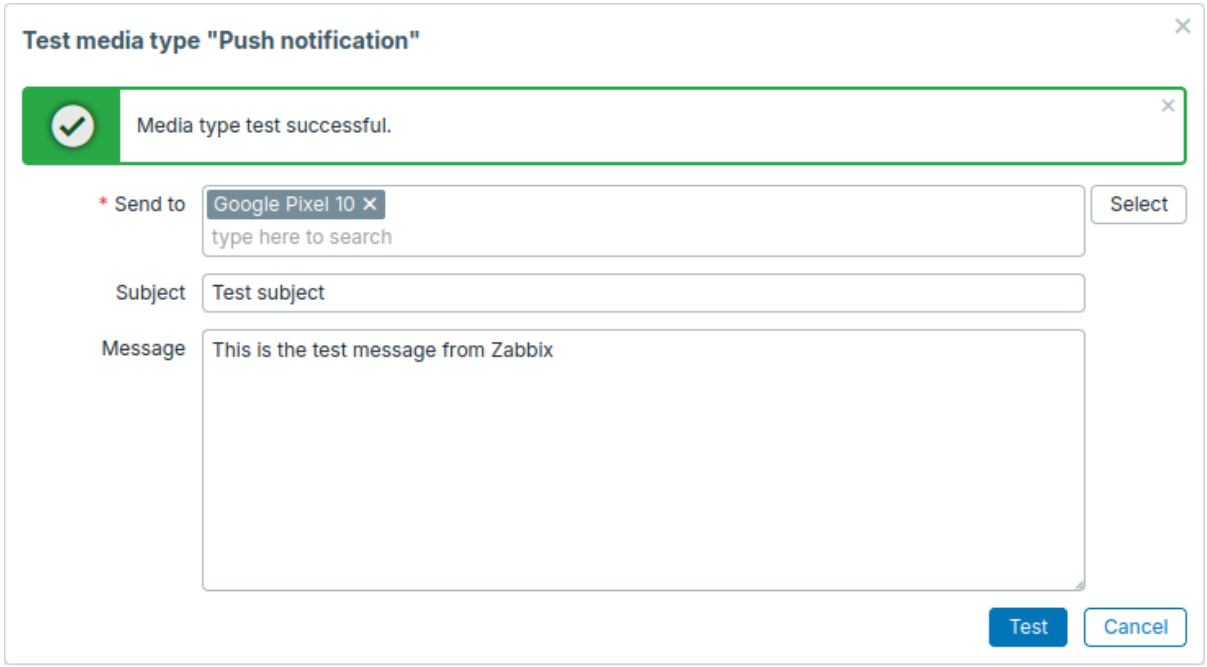
Po utworzeniu i włączeniu typu nośnika dodaj urządzenie użytkownika, aby automatycznie dodać typ nośnika do tego użytkownika (lub ręcznie dodaj nośnik użytkownika). Umożliwi to użytkownikowi odbieranie powiadomień push na urządzeniu mobilnym.

Testowanie

Aby sprawdzić, czy ten typ nośnika działa:

1. Znajdź typ nośnika na liście typów nośników.
2. Kliknij *Test* w ostatniej kolumnie listy, aby otworzyć formularz testowania.
3. W polu *Wyślij do* wybierz urządzenie.
4. Kliknij *Test*, aby wysłać testowe powiadomienie push do wybranego urządzenia.

W tym samym oknie pojawi się komunikat o powodzeniu lub niepowodzeniu:



SMS

Przegląd

Zabbix obsługuje wysyłanie wiadomości SMS przy użyciu szeregowego modemu GSM podłączonego do portu szeregowego serwera Zabbix.

Upewnij się, że:

- Prędkość urządzenia szeregowego (zwykle /dev/ttyS0 w systemie Linux) odpowiada prędkości modemu GSM. Zabbix nie ustawia prędkości połączenia szeregowego. Używa ustawień domyślnych.
- Użytkownik 'zabbix' ma uprawnienia do odczytu/zapisu urządzenia szeregowego. Uruchom polecenie ls -l /dev/ttyS0, aby sprawdzić bieżące uprawnienia urządzenia szeregowego.
- W modemie GSM wprowadzono kod PIN i jest on zachowywany po ponownym włączeniu zasilania. Alternatywnie można wyłączyć kod PIN na karcie SIM. Kod PIN można wprowadzić, wydając polecenie AT+CPIN="NNNN" (NNNN to Twój numer PIN, cudzysłowy muszą być obecne) w programie terminalowym, takim jak Unix minicom lub Windows HyperTerminal.

Zabbix został przetestowany z następującymi modemami GSM:

- Siemens MC35
- Teltonika ModemCOM/G10

Aby skonfigurować SMS jako kanał dostarczania wiadomości, należy również skonfigurować SMS jako typ nośnika oraz wprowadzić odpowiednie numery telefonów dla użytkowników.

Konfiguracja

Aby skonfigurować SMS jako typ nośnika:

1. Przejdź do *Alerts* > *Media types*.
2. Kliknij *Create media type* (lub kliknij *SMS* na liście wstępnie zdefiniowanych typów nośników).
3. Wprowadź parametry typu nośnika SMS w formularzu.

Następujący parametr jest specyficzny dla typu nośnika SMS:

Parameter	Description
GSM modem	Ustaw nazwę urządzenia szeregowego modemu GSM. Wprowadzona tutaj ścieżka zostanie zweryfikowana względem parametru serwera SMSDevices (jeśli został określony).

Szczegółowe informacje na temat konfigurowania domyślnych wiadomości i opcji przetwarzania alertów zawiera sekcja common media type parameters. Należy pamiętać, że równoległe przetwarzanie wysyłania powiadomień SMS nie jest możliwe.

Media użytkownika

Po skonfigurowaniu typu nośnika SMS przejdź do sekcji *Użytkownicy* > *Użytkownicy* i edytuj profil użytkownika, aby przypisać mu nośnik SMS. Czynności związane z konfigurowaniem mediów użytkownika, wspólne dla wszystkich typów nośników, opisano na stronie Typy nośników.

Niestandardowe skrypty alertów

Przegląd

Jeśli nie jesteś zadowolony z istniejących typów mediów do wysyłania alertów, istnieje alternatywny sposób ich obsługi. Możesz utworzyć skrypt, który będzie obsługiwał powiadomienie w wybrany przez Ciebie sposób.

Niestandardowe skrypty alertów są wykonywane na serwerze Zabbix. Skrypty te muszą znajdować się w katalogu określonym w parametrze AlertScriptsPath w pliku konfiguracji serwera.

Oto przykład niestandardowego skryptu alertu:

```
#####!/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 sprawdza kod zakończenia wykonanych poleceń i skryptów. Każdy kod zakończenia inny niż **0** jest uznawany za błąd wykonania polecenia. W takich przypadkach Zabbix spróbuje ponowić nieudaną próbę wykonania.

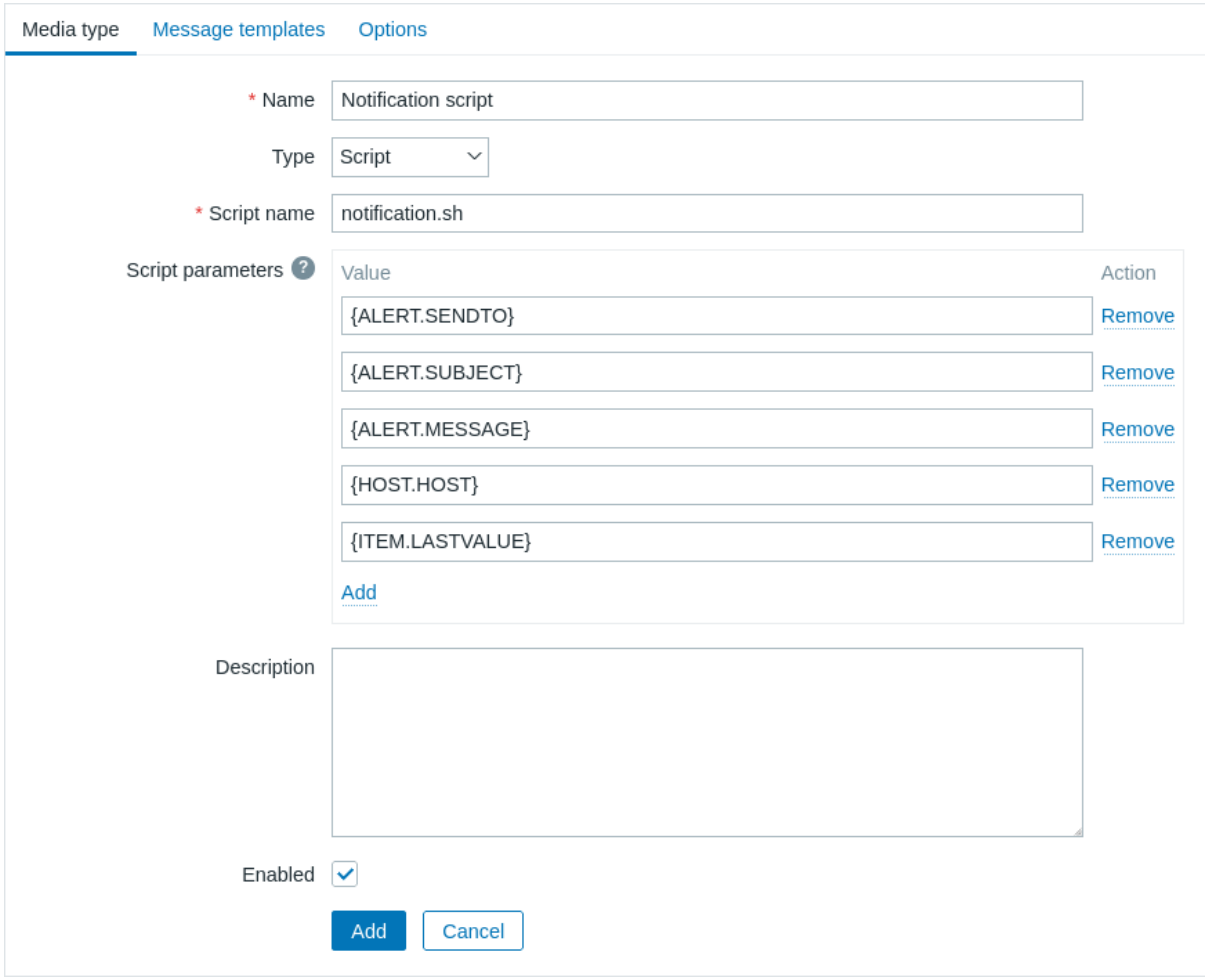
Zmienne środowiskowe nie są zachowywane ani tworzone dla skryptu, więc należy obsłużyć je jawnie.

Konfiguracja

Aby skonfigurować niestandardowe skrypty alertów jako typ nośnika:

1. Przejdź do *Alerty* > *Typy nośników*.
2. Kliknij *Utwórz typ nośnika*.
3. Wprowadź parametry typu nośnika skryptu w formularzu.

Karta **Typ nośnika** zawiera ogólne atrybuty typu nośnika:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Poniższe parametry są specyficzne dla typu nośnika skryptu:

Parametr	Opis
Nazwa skryptu	Wprowadź nazwę pliku skryptu (np. notification.sh), który znajduje się w katalogu określonym w parametrze konfiguracji serwera AlertScriptsPath.
Parametry skryptu	Dodaj opcjonalne parametry skryptu, które zostaną przekazane do skryptu jako argumenty wiersza poleceń w kolejności, w jakiej zostały zdefiniowane. Parametry skryptu obsługują makra {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} oraz wszystkie makra obsługiwane w powiadomieniach, a także makra użytkownika.

Szczegółowe informacje na temat konfigurowania domyślnych wiadomości i opcji przetwarzania alertów zawiera sekcja wspólne parametry typu nośnika.

Warning:
Nawet jeśli skrypt alertu nie używa domyślnych wiadomości, szablony wiadomości dla typów operacji używanych przez ten typ nośnika nadal muszą być zdefiniowane. W przeciwnym razie powiadomienie nie zostanie wysłane.

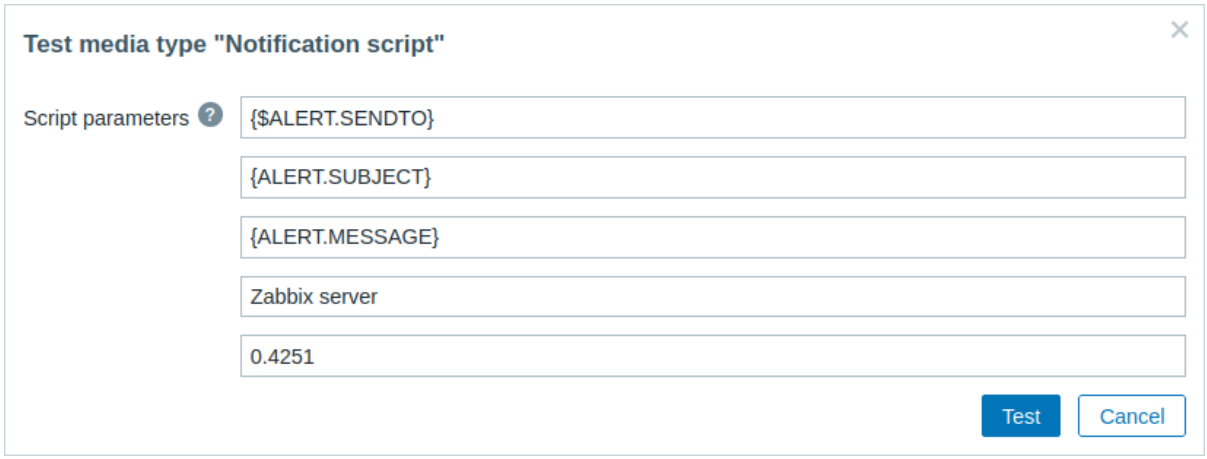
Attention:
Jeśli skonfigurowano więcej niż jeden typ nośnika skryptu, skrypty te mogą być przetwarzane równolegle przez procesy alertera. Łączna liczba procesów alertera jest ograniczona przez parametr `StartAlerters` w pliku konfiguracyjnym serwera.

Testowanie

Aby przetestować skonfigurowany typ mediów skryptu:

1. Znajdź odpowiedni skrypt na liście typów mediów.

2. Kliknij *Test* w ostatniej kolumnie listy; formularz testowy zostanie otwarty w oknie podręcznym. Formularz testowy będzie zawierał taką samą liczbę parametrów, jaka została skonfigurowana dla typu mediów skryptu.
3. W razie potrzeby edytuj wartości parametrów skryptu. Edycja wpływa wyłącznie na procedurę testową; rzeczywiste wartości nie zostaną zmienione.
4. Kliknij *Test*.



Note:
Podczas testowania skonfigurowanego typu mediów skryptu {ALERT.SENDTO}, {ALERT.SUBJECT}, {ALERT.MESSAGE} oraz makra użytkownika zostaną rozwinięte do swoich wartości, ale makra powiązane ze zdarzeniami (np. {HOST.HOST}, {ITEM.LASTVALUE} itd.) nie zostaną rozwinięte, ponieważ podczas testowania nie ma powiązanego zdarzenia, z którego można pobrać szczegóły. Należy pamiętać, że makra wewnątrz makr {ALERT.SUBJECT} i {ALERT.MESSAGE} również nie zostaną rozwinięte. Na przykład, jeśli wartość {ALERT.SUBJECT} jest złożona z "Problem: {EVENT.NAME}", to makro {EVENT.NAME} nie zostanie rozwinięte.

Media użytkownika

Po skonfigurowaniu typu mediów przejdź do sekcji *Users* > *Users* i edytuj profil użytkownika, przypisując ten typ mediów do użytkownika. Kroki konfiguracji mediów użytkownika, wspólne dla wszystkich typów mediów, są opisane na stronie Media types.

Należy pamiętać, że podczas definiowania mediów użytkownika pole *Send to* nie może być puste. Jeśli to pole nie jest używane w skrypcie alertu, wprowadź dowolną kombinację obsługiwanych znaków, aby ominąć wymagania walidacji.

Webhook

Przegląd

Typ nośnika webhook jest przydatny do wykonywania wywołań HTTP przy użyciu niestandardowego kodu JavaScript w celu łatwej integracji z zewnętrznym oprogramowaniem, takim jak systemy helpdesk, czaty lub komunikatory. Możesz wybrać import integracji dostarczonej przez Zabbix lub utworzyć własną integrację od podstaw.

Integracje

Dostępne są następujące integracje, które umożliwiają używanie predefiniowanych typów mediów webhook do wysyłania powiadomień Zabbix do:

- 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:
Oprócz wymienionych tutaj usług Zabbix można zintegrować z **Spiceworks** (webhook nie jest wymagany). Aby przekształcać powiadomienia Zabbix w zgłoszenia Spiceworks, utwórz typ mediów e-mail i w ustawieniach profilu wyznaczonego użytkownika Zabbix wprowadź adres e-mail działu pomocy Spiceworks (np. help@zabbix.on.spiceworks.com).

Konfiguracja

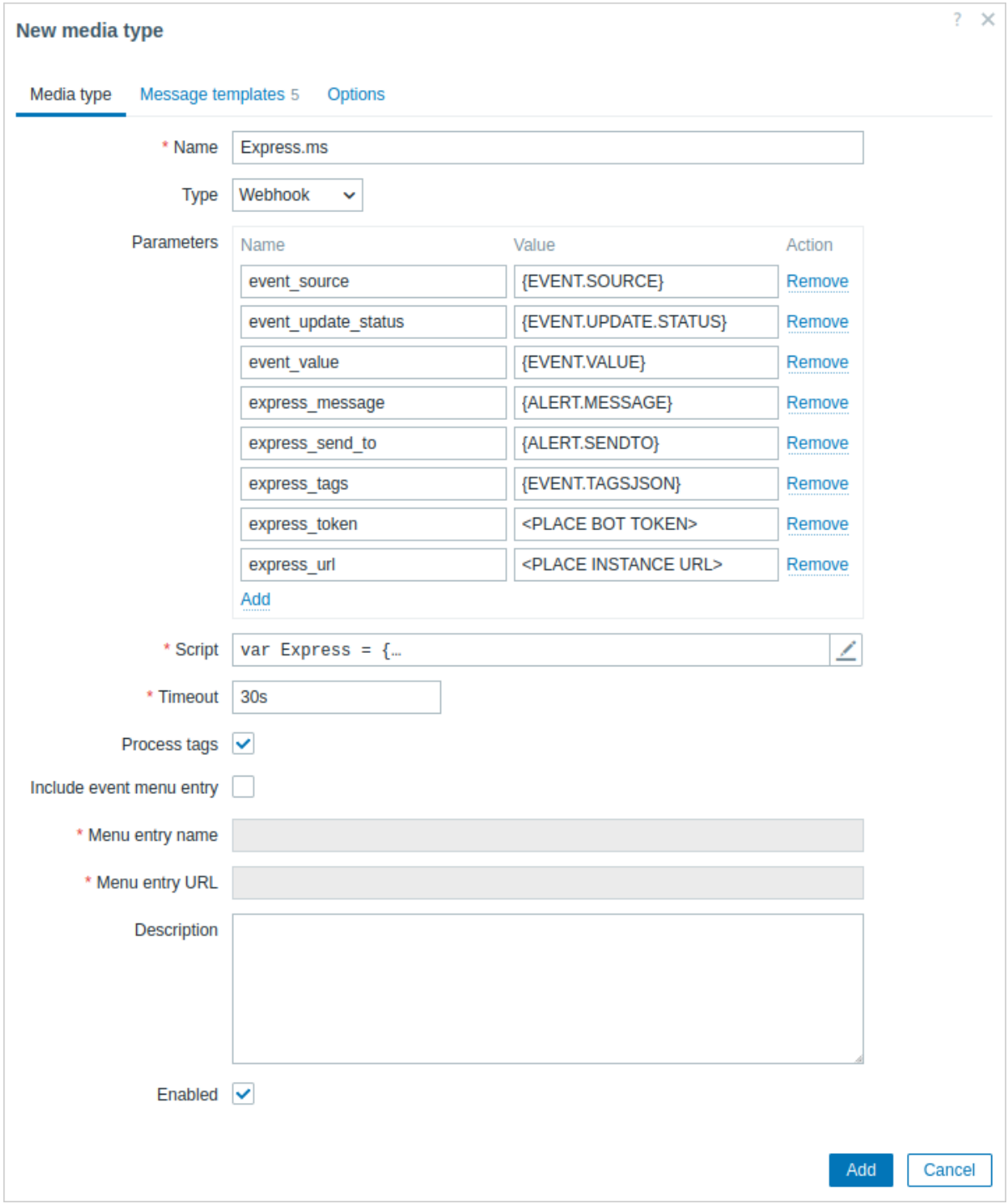
Aby rozpocząć korzystanie z integracji webhook:

1. Znajdź wymagany plik `.yaml` w katalogu `templates/media` pobranej wersji Zabbix lub pobierz go z repozytorium git Zabbix.
2. Zaimportuj plik do swojej instalacji Zabbix. Webhook pojawi się na liście typów mediów.
3. Skonfiguruj webhook zgodnie z instrukcjami w pliku `Readme.md` (możesz kliknąć nazwę webhook powyżej, aby szybko otworzyć plik `Readme.md`).

Aby utworzyć niestandardowy webhook od podstaw:

1. Przejdź do *Alerty* > *Typy mediów*.
2. Kliknij *Utwórz typ mediów*.
3. Wprowadź parametry typu mediów webhook w formularzu.

Karta **Typ mediów** zawiera różne atrybuty właściwe dla tego typu mediów:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Poniższe parametry są właściwe dla typu mediów webhook:

Parametr	Opis
Parametry	Określ zmienne webhook jako pary atrybut-wartość. W przypadku wstępnie skonfigurowanych webhook lista parametrów różni się w zależności od usługi. Opis parametrów znajdziesz w pliku *Readme.md* webhook. W przypadku nowych webhook domyślnie uwzględniono kilka typowych zmiennych (URL:<empty>, HTTPProxy:<empty>, To:{ALERT.SENDTO}, Subject:{ALERT.SUBJECT}, Message:{ALERT.MESSAGE}); możesz je pozostawić lub usunąć. Parametry webhook obsługują makra użytkownika, wszystkie makra obsługiwane w powiadomieniach o problemach, a dodatkowo makra {ALERT.SENDTO}, {ALERT.SUBJECT} i {ALERT.MESSAGE}. Jeśli określisz proxy HTTP, pole obsługuje taką samą funkcjonalność jak pole proxy HTTP w konfiguracji pozycji. Ciąg proxy może być poprzedzony przez `[scheme]://`, aby określić używany rodzaj proxy (np. https, socks4, socks5; zobacz dokumentację).
Skrypt	Wprowadź kod JavaScript w edytorze modalnym, który otwiera się po kliknięciu pola parametru lub ikony ołówka obok niego. Kod ten wykona operację webhook. Skrypt jest kodem funkcji, która przyjmuje pary parametr-wartość. Wartości należy konwertować na obiekty JSON za pomocą metody JSON.parse(), na przykład: `var params = JSON.parse(value);`. Kod ma dostęp do wszystkich parametrów, może wykonywać żądania HTTP GET, POST, PUT i DELETE, obsługiwać dodatkowe metody, takie jak CONNECT, PATCH, HEAD, OPTIONS i TRACE, a także zarządzać nagłówkami HTTP i treścią żądania. Skrypt musi zawierać operator return, w przeciwnym razie nie będzie prawidłowy. Może zwracać status OK wraz z opcjonalną listą tagów i wartości tagów (zobacz opcję *Przetwarzaj tagi*) albo ciąg błędu. Zdarzenia odtworzenia (niezależnie od tego, czy zostały wygenerowane automatycznie, czy w wyniku ręcznego zamknięcia) są tworzone przez serwer i zawierają rozwiązane tagi zdarzenia (w tym tagi odziedziczone z szablonów, hostów i wyzwalaczy). Skrypty webhook są wykonywane po utworzeniu alertu, dlatego tagi zwrócone przez skrypt webhook są dodawane dopiero po początkowym utworzeniu alertu i nie będą obecne w makrach {EVENT.TAGS} i `{EVENT.RECOVERY.TAGS}` początkowego komunikatu o problemie ani bezpośredniego komunikatu o odtworzeniu. *Uwaga*: Zaleca się używanie zmiennych lokalnych (np. `var local = 1`) zamiast zmiennych globalnych (np. `global = 1`), aby zapewnić, że każdy skrypt działa na własnych danych, oraz uniknąć kolizji między jednoczesnymi wywołaniami (zobacz znane problemy). Zobacz także: [Wytyczne dotyczące tworzenia webhook] (https://www.zabbix.com/documentation/guidelines/en/webhooks), Przykłady skryptów webhook, Dodatkowe obiekty JavaScript.
Limit czasu	Limit czasu wykonywania JavaScript (1-60 s, domyślnie 30 s). Obsługiwane są przyrostki czasu (np. `30s`, `1m`).
Przetwarzaj tagi	Zaznacz pole wyboru, aby przetwarzać wartości zwróconych właściwości JSON jako tagi. Tagi te są dodawane do wszystkich istniejących tagów problemu. Pamiętaj, że w przypadku używania tagów webhook webhook musi zwracać obiekt JSON zawierający co najmniej pusty obiekt tagów: `var result = {tags: {}};` Przykłady tagów, które mogą zostać zwrócone: *jira-id:prod-1234*, *responsible:John Smith*, *processed:<no value>*
Uwzględnij pozycję w menu zdarzenia	Zaznacz pole wyboru, aby uwzględnić pozycję w menu zdarzenia zawierającą odnośnik do utworzonego zewnętrznego zgłoszenia. Pozycja zostanie uwzględniona dla każdego włączonego webhook, dla którego zaznaczono to pole wyboru. Pamiętaj, że jeśli parametry *Nazwa pozycji menu* i *URL pozycji menu* zawierają makra {EVENT.TAGS.<tag name>}, pozycja zostanie uwzględniona tylko wtedy, gdy te makra mogą zostać rozwiązane (czyli gdy dla zdarzenia zdefiniowano te tagi). Jeśli opcja jest zaznaczona, webhook nie powinien być używany do wysyłania powiadomień do różnych użytkowników (rozważ utworzenie dedykowanego użytkownika) ani używany w wielu działaniach alertów dla pojedynczego zdarzenia problemu.
Nazwa pozycji menu	Określ nazwę pozycji menu. Obsługiwane jest makro {EVENT.TAGS.<tag name>}. To pole jest wymagane tylko wtedy, gdy zaznaczono opcję *Uwzględnij pozycję w menu zdarzenia*.

Parametr	Opis
URL pozycji menu	Określ docelowy URL pozycji menu. Obsługiwane jest makro {EVENT.TAGS.<tag name>}. To pole jest wymagane tylko wtedy, gdy zaznaczono opcję *Uwzględnij pozycję w menu zdarzenia*.

Szczegółowe informacje dotyczące konfigurowania domyślnych komunikatów i opcji przetwarzania alertów znajdziesz w sekcji wspólne parametry typu mediów.

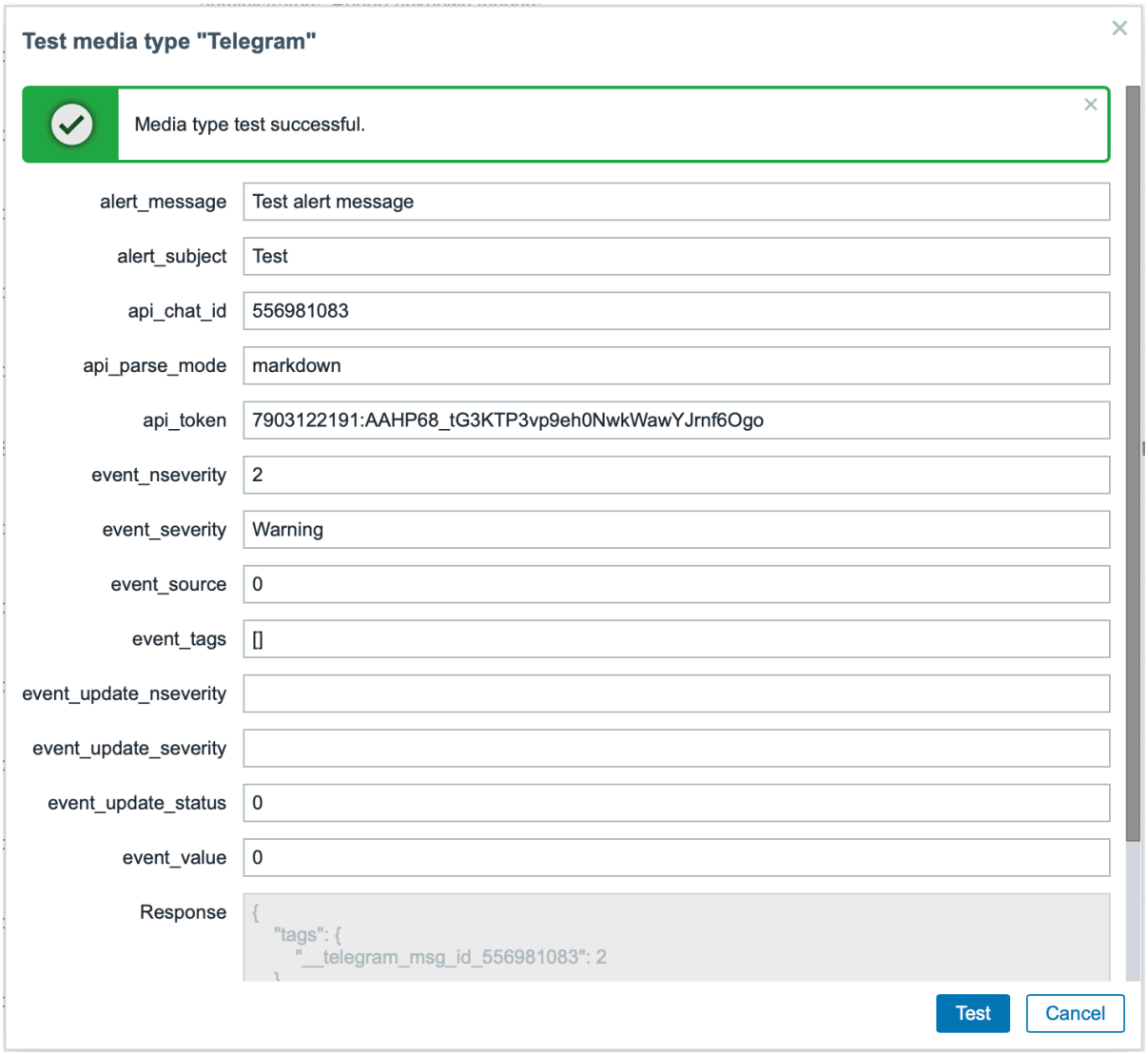
Warning:
Nawet jeśli webhook nie korzysta z domyślnych komunikatów, nadal należy zdefiniować szablony komunikatów dla typów operacji używanych przez ten webhook.

Testowanie

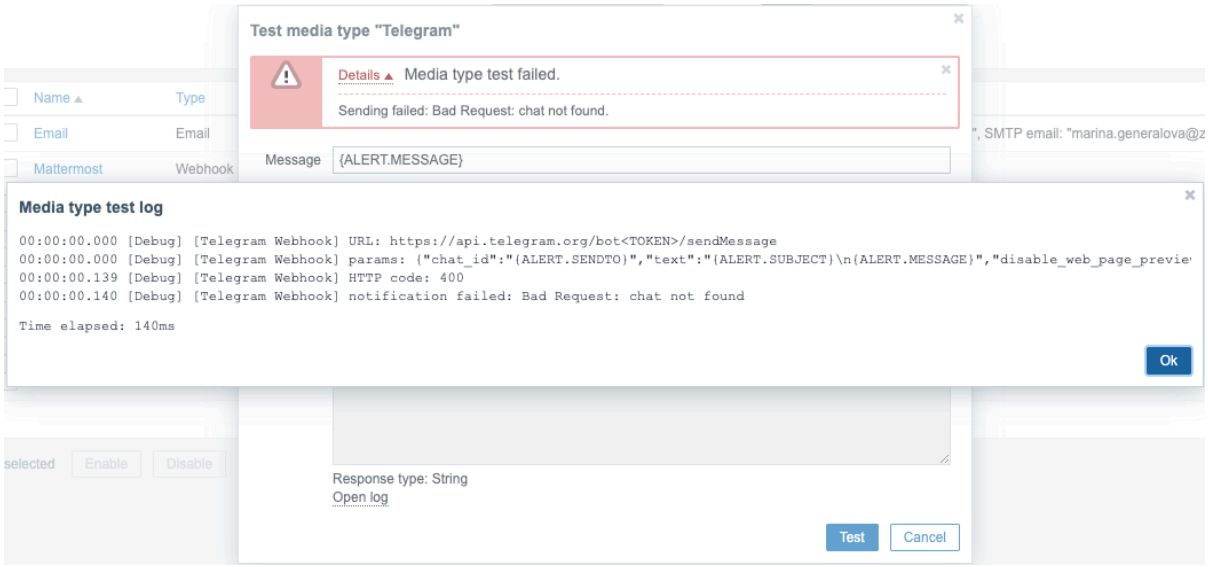
Aby przetestować skonfigurowany typ nośnika webhook:

1. Znajdź odpowiedni webhook na liście typów nośników.
2. Kliknij *Test* w ostatniej kolumnie listy (otworzy się okno testowania).
3. W razie potrzeby edytuj wartości parametrów webhooka.
   Zastąp makra przykładowymi wartościami. W przeciwnym razie makra nie zostaną rozwinięte, a test zakończy się niepowodzeniem.
4. Kliknij *Test*.

Zastępowanie lub usuwanie wartości w oknie testowania wpływa wyłącznie na procedurę testową; rzeczywiste wartości atrybutów webhooka pozostaną niezmienione.



Aby wyświetlić wpisy dziennika testu typu nośnika bez opuszczania okna testowania, kliknij *Open log* (otworzy się nowe okno podręczne).



Jeśli test webhooka zakończy się powodzeniem:

- Wyświetlany jest komunikat *„Media type test successful.”*.
- Odpowiedź serwera pojawia się w szarym polu *Response*.
- Typ odpowiedzi (JSON lub String) jest określony poniżej pola *Response*.

Jeśli test webhooka zakończy się niepowodzeniem:

- Wyświetlany jest komunikat *„Test typu nośnika nie powiódł się.”*, po którym podawane są dodatkowe szczegóły dotyczące błędu.

Media użytkownika

Po skonfigurowaniu typu nośnika przejdź do sekcji *Users* > *Users* i przypisz nośnik webhook do istniejącego użytkownika lub utwórz nowego użytkownika reprezentującego webhook.

Czynności związane z konfigurowaniem mediów użytkownika dla istniejącego użytkownika, wspólne dla wszystkich typów nośników, opisano na stronie Media types.

Jeśli webhook używa znaczników do przechowywania identyfikatora zgłoszenia\wiadomości, unikaj przypisywania tego samego webhooka jako nośnika do różnych użytkowników, ponieważ może to powodować błędy webhooka (dotyczy to większości webhooków korzystających z opcji *Include event menu entry*). W takim przypadku najlepszym rozwiązaniem jest utworzenie dedykowanego użytkownika reprezentującego webhook:

1. Po skonfigurowaniu typu nośnika webhook przejdź do sekcji *Users* > *Users* i utwórz dedykowanego użytkownika Zabbix reprezentującego webhook, na przykład użytkownika o nazwie *Slack* dla webhooka Slack. Wszystkie ustawienia poza nośnikiem można pozostawić domyślne, ponieważ ten użytkownik nie będzie logować się do Zabbix.
2. W profilu użytkownika przejdź do karty *Media* i dodaj webhook, podając wymagane dane kontaktowe. Jeśli webhook nie używa pola *Send to*, wprowadź dowolną kombinację obsługiwanych znaków, aby spełnić wymagania walidacji.
3. Przyznaj temu użytkownikowi co najmniej uprawnienia do odczytu uprawnień do wszystkich hostów, dla których powinien wysyłać alerty.

Podczas konfigurowania akcji alertu dodaj tego użytkownika w polu *Send to users* w sekcji Operation details — spowoduje to, że Zabbix będzie używać webhooka do wysyłania powiadomień z tej akcji.

Konfigurowanie akcji alertów

Akcje określają, które powiadomienia powinny być wysyłane za pośrednictwem webhooka. Kroki dotyczące konfigurowania akcji z użyciem webhooków są takie same jak dla wszystkich innych typów mediów, z następującymi wyjątkami:

- Jeśli webhook używa tagów webhooka do przechowywania identyfikatora zgłoszenia/wiadomości i obsługi operacji aktualizacji/rozwiązania, unikaj używania tego samego webhooka w wielu akcjach alertów dla pojedynczego zdarzenia problemu. Jeśli {EVENT.TAGS.<tag name>} istnieje i zostanie zaktualizowany w webhooku, jego wynikowa wartość będzie niezdefiniowana. Aby tego uniknąć, użyj nowej nazwy tagu w webhooku do przechowywania zaktualizowanych wartości. Dotyczy to webhooków Jira, Jira Service Desk, Mattermost, Opsgenie, OTRS, Redmine, ServiceNow, Slack, Zammad i Zendesk dostarczanych przez Zabbix oraz większości webhooków wykorzystujących opcję *Include event menu entry*. Należy jednak pamiętać, że pojedynczy webhook może być używany w wielu operacjach lub krokach eskalacji tej samej akcji, a także w różnych akcjach, które nie zostaną wywołane przez to samo zdarzenie problemu z powodu różnych warunków.
- Podczas używania webhooka w akcjach dla zdarzeń wewnętrznych upewnij się, że zaznaczono pole *Custom message* i zdefiniowano niestandardową wiadomość w konfiguracji operacji akcji. W przeciwnym razie powiadomienie nie zostanie wysłane.

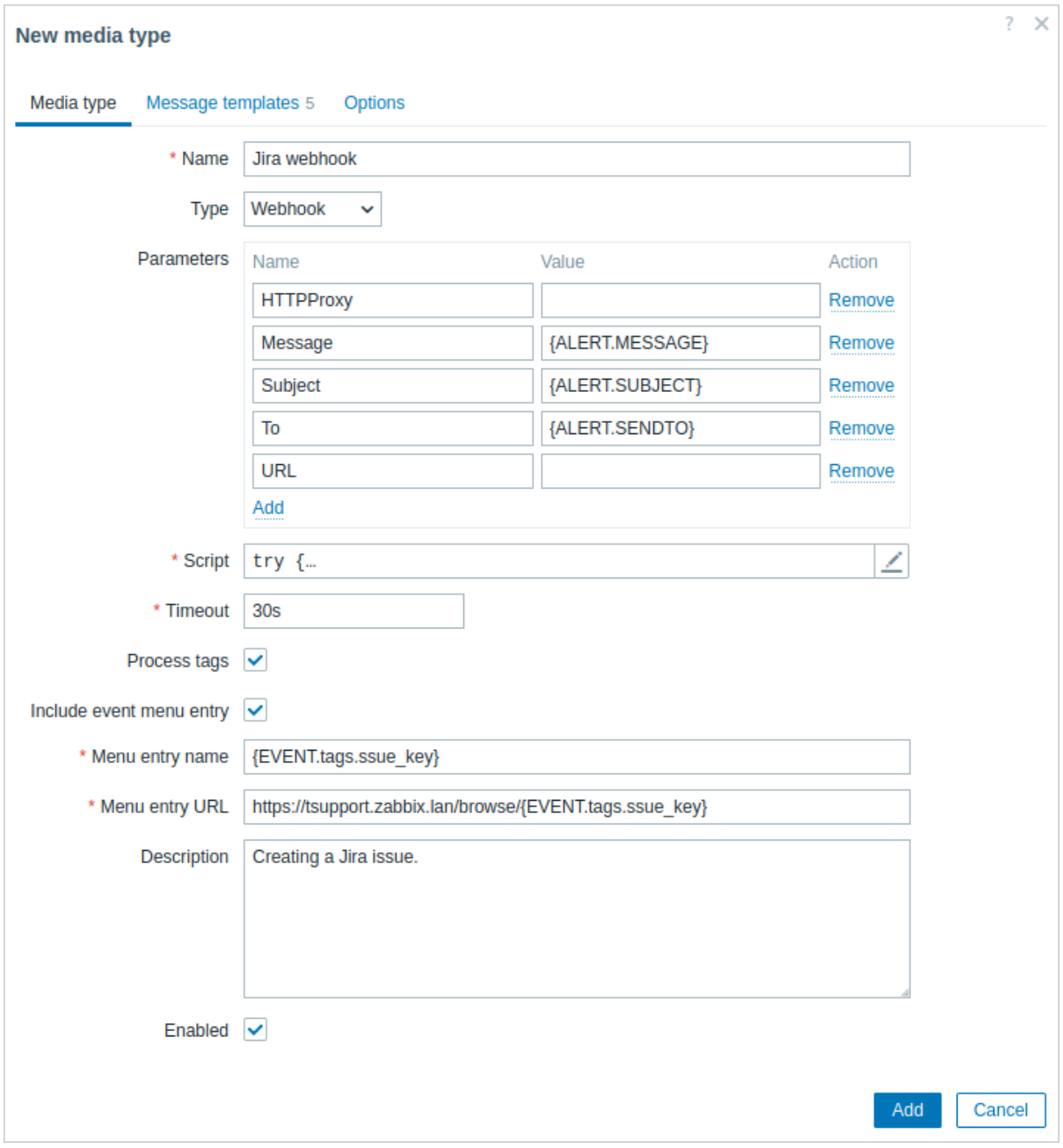
Przegląd

Chociaż Zabbix oferuje dużą liczbę integracji webhook dostępnych od razu po instalacji, możesz zamiast tego chcieć utworzyć własne webhooki. Ta sekcja zawiera przykłady niestandardowych skryptów webhook (używanych w parametrze *Script*). Zobacz sekcję webhook, aby uzyskać opis innych parametrów webhook.

Attention:
Nie używaj niezadeklarowanych przypisań w JavaScript preprocesingu. Do deklarowania zmiennych lokalnych używaj `var`.

Zdarzenia odzyskania (niezależnie od tego, czy zostały wygenerowane automatycznie, czy po ręcznym zamknięciu) zawierają tagi odzyskanego zdarzenia (w tym tagi odziedziczone z szablonów, hostów i wyzwalaczy). Skrypty webhook są wykonywane po utworzeniu alertu; tagi zwracane przez skrypt webhook są więc stosowane po początkowym utworzeniu alertu i mogą nie być obecne w alercie, który wywołał webhook. Jeśli integracja wymaga, aby tagi wygenerowane przez webhook były obecne w początkowym powiadomieniu, pobierz tagi zdarzenia z serwera (na przykład za pomocą API Event) albo przechowuj tagi wygenerowane przez webhook w zewnętrznym trwałym magazynie i wykonuj tam korelację.

Jira webhook (niestandardowy)



Ten skrypt utworzy zgłoszenie JIRA i zwróci pewne informacje o utworzonym zgłoszeniu.

```
try {
    Zabbix.log(4, '[ Jira webhook ] Started with params: ' + 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 'Response code: ' + req.getStatus();
    }

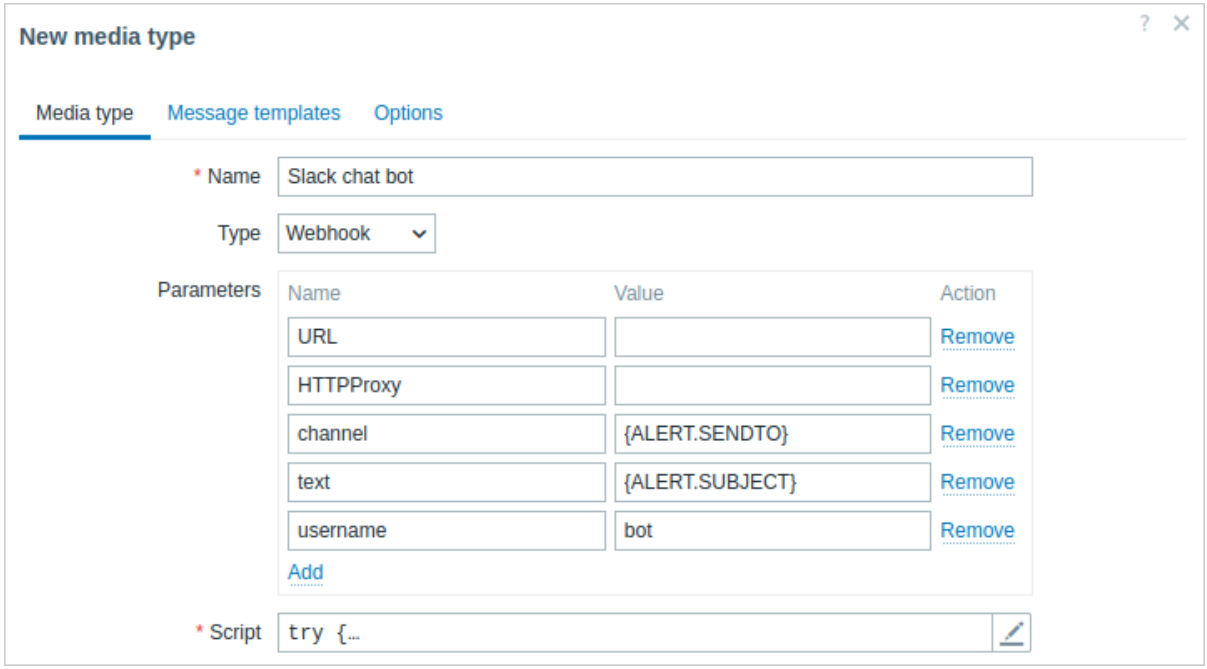
    resp = JSON.parse(resp);
    result.tags.issue_id = resp.id;
    result.tags.issue_key = resp.key;

    return JSON.stringify(result);
}
catch (error) {
    Zabbix.log(4, '[ Jira webhook ] Issue creation failed json : ' + JSON.stringify({"fields": fields}));
    Zabbix.log(3, '[ Jira webhook ] issue creation failed : ' + error);

    throw 'Failed with error: ' + error;
}
```

Slack webhook (niestandardowy)

Ten webhook przekazuje powiadomienia z Zabbix do kanału 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;
}
```

Akcje

Przegląd

Jeśli chcesz, aby w wyniku zdarzeń były wykonywane określone operacje (na przykład wysyłane powiadomienia), musisz skonfigurować akcje.

Akcje mogą być definiowane w odpowiedzi na zdarzenia wszystkich obsługiwanych typów:

- Akcje wyzwalaczy - dla zdarzeń, gdy stan wyzwalacza zmienia się między `OK` a `Problem`
- Akcje usług - dla zdarzeń, gdy stan usługi zmienia się między `OK` a `Problem`
- Akcje wykrywania - dla zdarzeń, gdy odbywa się wykrywanie sieci
- Akcje autorejestracji - dla zdarzeń, gdy nowe aktywne agenty rejestrują się automatycznie (lub gdy metadane hosta zmieniają się dla już zarejestrowanych)
- Akcje wewnętrzne - dla zdarzeń, gdy pozycje stają się nieobsługiwane lub wyzwalacze przechodzą w stan nieznany

Kluczowe różnice dotyczące akcji usług to:

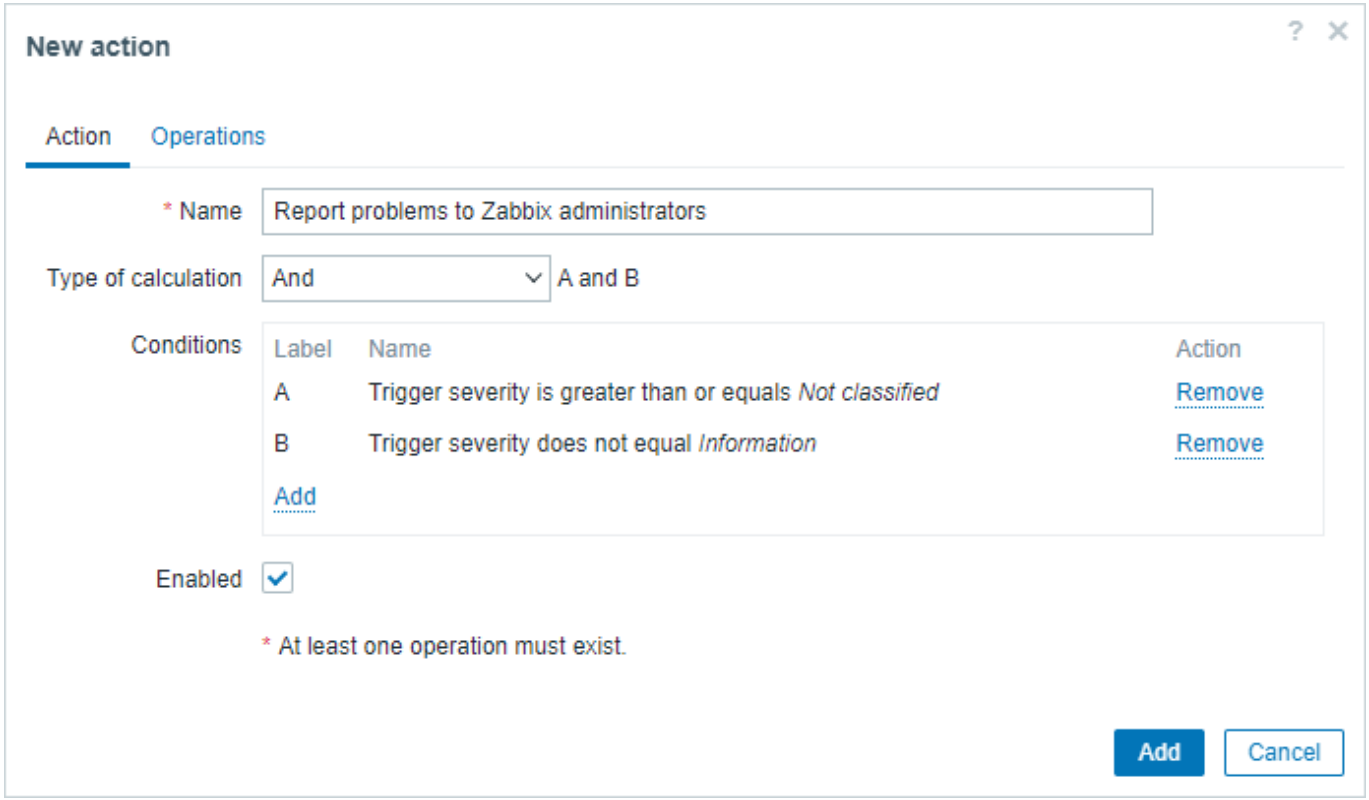
- Dostęp użytkownika do akcji usług zależy od uprawnień dostępu do usług przyznanych przez rolę
- Akcje usług obsługują inny zestaw warunków

Konfigurowanie akcji

Aby skonfigurować akcję, wykonaj następujące czynności:

1. Przejdź do *Alerts* > *Actions* i wybierz wymagany typ akcji z podmenu (możesz później przełączyć się na inny typ, używając listy rozwijanej w tytule).
2. Kliknij *Create action*.
3. Nadaj nazwę akcji.
4. Wybierz warunki, na podstawie których będą wykonywane operacje.
5. Wybierz operacje, które mają zostać wykonane.

Ogólne atrybuty akcji:



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Unikalna nazwa akcji.

Parameter	Description
Type of calculation	Wybierz opcję oceny warunków akcji (gdy jest więcej niż jeden warunek): **And** - wszystkie warunki muszą być spełnione. **Or** - wystarczy, że spełniony jest jeden warunek. **And/Or** - połączenie obu: AND dla różnych typów warunków oraz OR dla warunków tego samego typu. **Custom expression** - zdefiniowana przez użytkownika formuła obliczeniowa do oceny warunków akcji.
Conditions	Lista warunków akcji. Kliknij *Add*, aby dodać nowy warunek. Jeśli nie skonfigurowano żadnych warunków, akcja zostanie uruchomiona dla każdego zdarzenia odpowiadającego konfigurowanemu typowi akcji.
Enabled	Zaznacz pole wyboru, aby włączyć akcję. W przeciwnym razie będzie ona wyłączona.

Warunki

Przegląd

Można zdefiniować, że akcja zostanie wykonana tylko wtedy, gdy zdarzenie spełnia określony zestaw warunków. Warunki ustawia się podczas konfigurowania action.

Dopasowanie warunków uwzględnia wielkość liter.

Akcje wyzwalaczy

W akcjach opartych na wyzwalaczach można używać następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
Grupa hostów	równa się nie równa się	Określ grupy hostów lub grupy hostów do wykluczenia. **równa się** - zdarzenie należy do tej grupy hostów. **nie równa się** - zdarzenie nie należy do tej grupy hostów. Określenie nadrzędnej grupy hostów niejawnie wybiera wszystkie zagnieżdżone grupy hostów. Aby wskazać tylko grupę nadrzędną, wszystkie zagnieżdżone grupy muszą zostać dodatkowo ustawione z operatorem **nie równa się**.
Szablon	równa się nie równa się	Określ szablony lub szablony do wykluczenia. **równa się** - zdarzenie należy do wyzwalacza dziedziczonego z tego szablonu. **nie równa się** - zdarzenie nie należy do wyzwalacza dziedziczonego z tego szablonu.
Host	równa się nie równa się	Określ hosty lub hosty do wykluczenia. **równa się** - zdarzenie należy do tego hosta. **nie równa się** - zdarzenie nie należy do tego hosta.
Nazwa tagu	równa się nie równa się zawiera nie zawiera	Określ tag zdarzenia lub tag zdarzenia do wykluczenia. **równa się** - zdarzenie ma ten tag. **nie równa się** - zdarzenie nie ma tego tagu. **zawiera** - zdarzenie ma tag zawierający ten ciąg znaków. **nie zawiera** - zdarzenie nie ma tagu zawierającego ten ciąg znaków.
Wartość tagu	równa się nie równa się zawiera nie zawiera	Określ kombinację tagu zdarzenia i wartości lub kombinację tagu i wartości do wykluczenia. **równa się** - zdarzenie ma ten tag i wartość. **nie równa się** - zdarzenie nie ma tego tagu i wartości. **zawiera** - zdarzenie ma tag i wartość zawierające te ciągi znaków. **nie zawiera** - zdarzenie nie ma tagu i wartości zawierających te ciągi znaków.
Wyzwalacz	równa się nie równa się	Określ wyzwalacze lub wyzwalacze do wykluczenia. **równa się** - zdarzenie jest generowane przez ten wyzwalacz. **nie równa się** - zdarzenie jest generowane przez dowolny inny wyzwalacz, z wyjątkiem tego.

Typ warunku	Obsługiwane operatory	Opis
Nazwa zdarzenia	zawiera nie zawiera	Określ ciąg znaków w nazwie zdarzenia wygenerowanego przez wyzwalacz lub ciąg znaków do wykluczenia. Domyślnie nazwa zdarzenia odpowiada nazwie wyzwalacza, chyba że w konfiguracji wyzwalacza określono niestandardową nazwę zdarzenia. **zawiera** - nazwa zdarzenia zawiera ten ciąg znaków. **nie zawiera** - ten ciąg znaków jest wykluczony z nazwy zdarzenia. Uwaga: Wprowadzona wartość będzie porównywana z nazwą zdarzenia po rozwinięciu wszystkich makr.
Poziom ważności wyzwalacza	równa się nie równa się jest większy lub równy jest mniejszy lub równy	Określ poziom ważności wyzwalacza. **równa się** - równy poziomowi ważności wyzwalacza. **nie równa się** - nie równy poziomowi ważności wyzwalacza. **jest większy lub równy** - większy lub równy poziomowi ważności wyzwalacza. **jest mniejszy lub równy** - mniejszy lub równy poziomowi ważności wyzwalacza.
Okres czasu	w nie w	Określ okres czasu lub okres czasu do wykluczenia. **w** - czas zdarzenia mieści się w okresie czasu. **nie w** - czas zdarzenia nie mieści się w okresie czasu. Opis formatu znajduje się na stronie specyfikacji okresu czasu. Obsługiwane są makra użytkownika.
Problem jest wyciszony	nie tak	Określ, czy problem jest wyciszony (niewyświetlany) z powodu konserwacji hosta. **nie** - problem nie jest wyciszony. **tak** - problem jest wyciszony.

Akcje usług

W akcjach usług można używać następujących warunków:

Condition type	Supported operators	Description
Service	equals does not equal	Określ usługę lub usługę do wykluczenia. **equals** - zdarzenie należy do tej usługi. **does not equal** - zdarzenie nie należy do tej usługi. Określenie usługi nadrzędnej automatycznie wybiera wszystkie usługi podrzędne. Aby wskazać wyłącznie usługę nadrzędną, wszystkie zagnieżdżone usługi trzeba dodatkowo ustawić operatorem **does not equal**.
Service name	contains does not contain	Określ ciąg znaków w nazwie usługi lub ciąg do wykluczenia. **contains** - zdarzenie jest generowane przez usługę zawierającą ten ciąg w nazwie. **does not contain** - tego ciągu nie można znaleźć w nazwie usługi.
Service tag name	equals does not equal contains does not contain	Określ tag zdarzenia lub tag zdarzenia do wykluczenia. Tagi zdarzeń usług można zdefiniować w sekcji konfiguracji usługi *Tags*. **equals** - zdarzenie ma ten tag. **does not equal** - zdarzenie nie ma tego tagu. **contains** - zdarzenie ma tag zawierający ten ciąg. **does not contain** - zdarzenie nie ma tagu zawierającego ten ciąg.
Service tag value	equals does not equal contains does not contain	Określ kombinację tagu zdarzenia i wartości lub kombinację tagu i wartości do wykluczenia. Tagi zdarzeń usług można zdefiniować w sekcji konfiguracji usługi *Tags*. **equals** - zdarzenie ma ten tag i wartość. **does not equal** - zdarzenie nie ma tego tagu i wartości. **contains** - zdarzenie ma tag i wartość zawierające te ciągi. **does not contain** - zdarzenie nie ma tagu i wartości zawierających te ciągi.

Attention:
Upewnij się, że dla akcji usług zdefiniowano szablony wiadomości w menu *Alerts* > *Media types*. W przeciwnym razie powiadomienia nie zostaną wysłane.

Działania wykrywania

W przypadku zdarzeń opartych na wykrywaniu można użyć następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
Adres IP hosta	równa się nie równa się	Określ zakres adresów IP lub zakres wykluczany dla wykrytego hosta. **równa się** - adres IP hosta znajduje się w zakresie. **nie równa się** - adres IP hosta nie znajduje się w zakresie. Można użyć następujących formatów: Pojedynczy adres IP: 192.168.1.33 Zakres adresów IP: 192.168.1-10.1-254 Maska 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 Format listy może zawierać spacje.
Typ usługi	równa się nie równa się	Określ typ wykrytej usługi lub typ usługi, który ma zostać wykluczony. **równa się** - pasuje do wykrytej usługi. **nie równa się** - nie pasuje do wykrytej usługi. Dostępne typy usług: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Zabbix agent, SNMPv1 agent, SNMPv2 agent, SNMPv3 agent, ICMP ping, telnet.
Port usługi	równa się nie równa się	Określ zakres portów TCP wykrytej usługi lub zakres, który ma zostać wykluczony. **równa się** - port usługi znajduje się w zakresie. **nie równa się** - port usługi nie znajduje się w zakresie.
Reguła wykrywania	równa się nie równa się	Określ regułę wykrywania lub regułę wykrywania, która ma zostać wykluczona. **równa się** - używana jest ta reguła wykrywania. **nie równa się** - używana jest dowolna inna reguła wykrywania z wyjątkiem tej reguły.
Sprawdzenie wykrywania	równa się nie równa się	Określ sprawdzenie wykrywania lub sprawdzenie wykrywania, które ma zostać wykluczone. **równa się** - używane jest to sprawdzenie wykrywania. **nie równa się** - używane jest dowolne inne sprawdzenie wykrywania z wyjątkiem tego sprawdzenia.
Obiekt wykrywania	równa się	Określ wykryty obiekt. **równa się** - jest równy wykrytemu obiektowi (urządzeniu lub usłudze).
Stan wykrywania	równa się	**Up** - pasuje do zdarzeń „Host Up” i „Service Up”. **Down** - pasuje do zdarzeń „Host Down” i „Service Down”. **Discovered** - pasuje do zdarzeń „Host Discovered” i „Service Discovered”. **Lost** - pasuje do zdarzeń „Host Lost” i „Service Lost”.
Czas działania/czas niedostępności	jest większy lub równy jest mniejszy lub równy	Czas działania dla zdarzeń „Host Up” i „Service Up”. Czas niedostępności dla zdarzeń „Host Down” i „Service Down”. **jest większy lub równy** - jest większy lub równy podanej wartości. Parametr jest podawany w sekundach. **jest mniejszy lub równy** - jest mniejszy lub równy podanej wartości. Parametr jest podawany w sekundach.

Typ warunku	Obsługiwane operatory	Opis
Odebrana wartość	równa się nie równa się jest większa lub równa jest mniejsza lub równa zawiera nie zawiera	Określ wartość odebraną ze sprawdzenia agent (Zabbix, SNMP) w regule wykrywania. Porównywane są ciągi znaków. Jeśli dla reguły skonfigurowano kilka sprawdzeń Zabbix agent lub SNMP, sprawdzane są odebrane wartości dla każdego z nich (każde sprawdzenie generuje nowe zdarzenie, które jest porównywane ze wszystkimi warunkami). **równa się** - jest równa wartości. **nie równa się** - nie jest równa wartości. **jest większa lub równa** - jest większa lub równa wartości. **jest mniejsza lub równa** - jest mniejsza lub równa wartości. **zawiera** - zawiera podciąg. Parametr jest podawany jako ciąg znaków. **nie zawiera** - nie zawiera podciągu. Parametr jest podawany jako ciąg znaków.
Proxy	równa się nie równa się	Określ proxy lub proxy, które ma zostać wykluczone. **równa się** - używane jest to proxy. **nie równa się** - używane jest dowolne inne proxy z wyjątkiem tego proxy.
Grupa proxy	równa się nie równa się	Określ grupę proxy lub grupę proxy, która ma zostać wykluczona. **równa się** - używana jest ta grupa proxy. **nie równa się** - używana jest dowolna inna grupa proxy z wyjątkiem tej grupy.

Note:
Sprawdzanie usług w regule wykrywania, którego wynikiem są zdarzenia wykrywania, nie odbywa się jednocześnie. Dlatego jeśli w działaniu skonfigurowano **wiele** wartości dla warunków `Typ usługi`, `Port usługi` lub `Odebrana wartość`, będą one porównywane z jednym zdarzeniem wykrywania naraz, ale **nie** z kilkoma zdarzeniami jednocześnie. W rezultacie działania z wieloma wartościami dla tych samych typów sprawdzeń mogą nie zostać wykonane prawidłowo.

Działania automatycznej rejestracji

W działaniach opartych na automatycznej rejestracji aktywnego agenta można używać następujących warunków:

Typ warunku	Obsługiwane operatory	Opis
Metadane hosta	zawiera nie zawiera pasuje nie pasuje	Określ metadane hosta lub metadane hosta do wykluczenia. **zawiera** - metadane hosta zawierają ciąg znaków. **nie zawiera** - metadane hosta nie zawierają ciągu znaków. Metadane hosta można określić w pliku konfiguracyjnym agenta. **pasuje** - metadane hosta pasują do wyrażenia regularnego. **nie pasuje** - metadane hosta nie pasują do wyrażenia regularnego.
Nazwa hosta	zawiera nie zawiera pasuje nie pasuje	Określ nazwę hosta lub nazwę hosta do wykluczenia. **zawiera** - nazwa hosta zawiera ciąg znaków. **nie zawiera** - nazwa hosta nie zawiera ciągu znaków. **pasuje** - nazwa hosta pasuje do wyrażenia regularnego. **nie pasuje** - nazwa hosta nie pasuje do wyrażenia regularnego.
Proxy	równa się nie równa się	Określ proxy lub proxy do wykluczenia. **równa się** - używane jest to proxy. **nie równa się** - używane jest dowolne inne proxy (lub serwer) z wyjątkiem tego proxy.
Grupa proxy	równa się nie równa się	Określ grupę proxy lub grupę proxy do wykluczenia. **równa się** - używana jest ta grupa proxy. **nie równa się** - używana jest dowolna inna grupa proxy z wyjątkiem tej grupy.

Akcje zdarzeń wewnętrznych

Dla akcji opartych na zdarzeniach wewnętrznych można ustawić następujące warunki:

Typ warunku	Obsługiwane operatory	Opis
Typ zdarzenia	equals	**Pozycja w stanie „nieobsługiwane”** - pasuje do zdarzeń, w których pozycja przechodzi ze stanu „normalny” do stanu „nieobsługiwane”. **Reguła wykrywania niskiego poziomu w stanie „nieobsługiwane”** - pasuje do zdarzeń, w których reguła wykrywania niskiego poziomu przechodzi ze stanu „normalny” do stanu „nieobsługiwane”. **Wyzwalacz w stanie „nieznany”** - pasuje do zdarzeń, w których wyzwalacz przechodzi ze stanu „normalny” do stanu „nieznany”.
Grupa hostów	equals does not equal	Określ grupy hostów lub grupy hostów do wykluczenia. **equals** - zdarzenie należy do tej grupy hostów. **does not equal** - zdarzenie nie należy do tej grupy hostów.
Nazwa tagu	equals does not equal contains does not contain	Określ tag zdarzenia lub tag zdarzenia do wykluczenia. **equals** - zdarzenie ma ten tag. **does not equal** - zdarzenie nie ma tego tagu. **contains** - zdarzenie ma tag zawierający ten ciąg znaków. **does not contain** - zdarzenie nie ma tagu zawierającego tego ciągu znaków.
Wartość tagu	equals does not equal contains does not contain	Określ kombinację tagu zdarzenia i wartości lub kombinację tagu i wartości do wykluczenia. **equals** - zdarzenie ma ten tag i wartość. **does not equal** - zdarzenie nie ma tego tagu i wartości. **contains** - zdarzenie ma tag i wartość zawierające te ciągi znaków. **does not contain** - zdarzenie nie ma tagu i wartości zawierających tych ciągów znaków.
Szablon	equals does not equal	Określ szablony lub szablony do wykluczenia. **equals** - zdarzenie należy do pozycji/wyzwalacza/reguły wykrywania niskiego poziomu dziedziczonych z tego szablonu. **does not equal** - zdarzenie nie należy do pozycji/wyzwalacza/reguły wykrywania niskiego poziomu dziedziczonych z tego szablonu.
Host	equals does not equal	Określ hosty lub hosty do wykluczenia. **equals** - zdarzenie należy do tego hosta. **does not equal** - zdarzenie nie należy do tego hosta.

Typ obliczania

Dostępne są następujące opcje obliczania warunków:

- **And** - wszystkie warunki muszą być spełnione

Zwróć uwagę, że użycie obliczania „And” jest niedozwolone między kilkoma wyzwalaczami, gdy są one wybrane jako warunek `Trigger=`. Akcje mogą być wykonywane wyłącznie na podstawie zdarzenia jednego wyzwalacza.

- **Or** - wystarczy, że jeden warunek jest spełniony
- **And/Or** - połączenie obu: AND dla różnych typów warunków oraz OR dla tego samego typu warunku, na przykład:

Grupa hostów równa się Oracle servers
 Grupa hostów równa się MySQL servers
 Nazwa zdarzenia zawiera ’Database is down’
 Nazwa zdarzenia zawiera ’Database is unavailable’

jest oceniane jako

(Grupa hostów równa się Oracle servers **or** Grupa hostów równa się MySQL servers**) and (**Nazwa zdarzenia zawiera ’Database is down’ **or** Nazwa zdarzenia zawiera ’Database is unavailable’**)**

- **Custom expression** - formuła obliczania zdefiniowana przez użytkownika do oceny warunków akcji. Musi zawierać wszystkie warunki (reprezentowane jako wielkie litery A, B, C, ...) i może zawierać spacje, tabulatory, nawiasy ( ), **and** (z rozróżnianiem wielkości liter), **or** (z rozróżnianiem wielkości liter), **not** (z rozróżnianiem wielkości liter).

Podczas gdy poprzedni przykład z `And/Or` byłby przedstawiony jako (A or B) and (C or D), w wyrażeniu niestandardowym możesz mieć również wiele innych sposobów obliczania:

(A and B) and (C or D)
 (A and B) or (C and D)
 ((A or B) and C) or D
 (not (A or B) and C) or not D
 itd.

Usuwanie obiektów używanych w akcjach

Podczas próby usunięcia obiektów, które są używane w warunku/operacji akcji, użytkownik otrzyma odpowiedni komunikat o błędzie.

Akcje nie są wyłączane, jeśli warunki lub operacje akcji:

- mają usunięte obiekty;
- mają obiekty, które nie są już wykrywane przez wykrywanie niskopoziomowe i zostały usunięte (po upływie okresu *Delete lost resources*).

Warunki/operacje akcji nie są usuwane, jeśli mają brakujące obiekty. Zamiast tego usunięte obiekty są wyświetlane jako *Deleted host group*, *Deleted host*, *Deleted trigger* itd. Ikona ostrzeżenia z komunikatem „This action has conditions or operations referencing deleted object(s).” jest wyświetlana w kolumnie „Info” na liście akcji.

Operacje

Przegląd

Dla wszystkich zdarzeń można zdefiniować następujące operacje:

- Wyślij wiadomość
- Wykonaj zdalne polecenie

Attention:
Serwer Zabbix nie tworzy alertów, jeśli dostęp do hosta jest jawnie „zabroniony” dla użytkownika zdefiniowanego jako odbiorca operacji akcji lub jeśli użytkownik w ogóle nie ma zdefiniowanych praw do hosta.

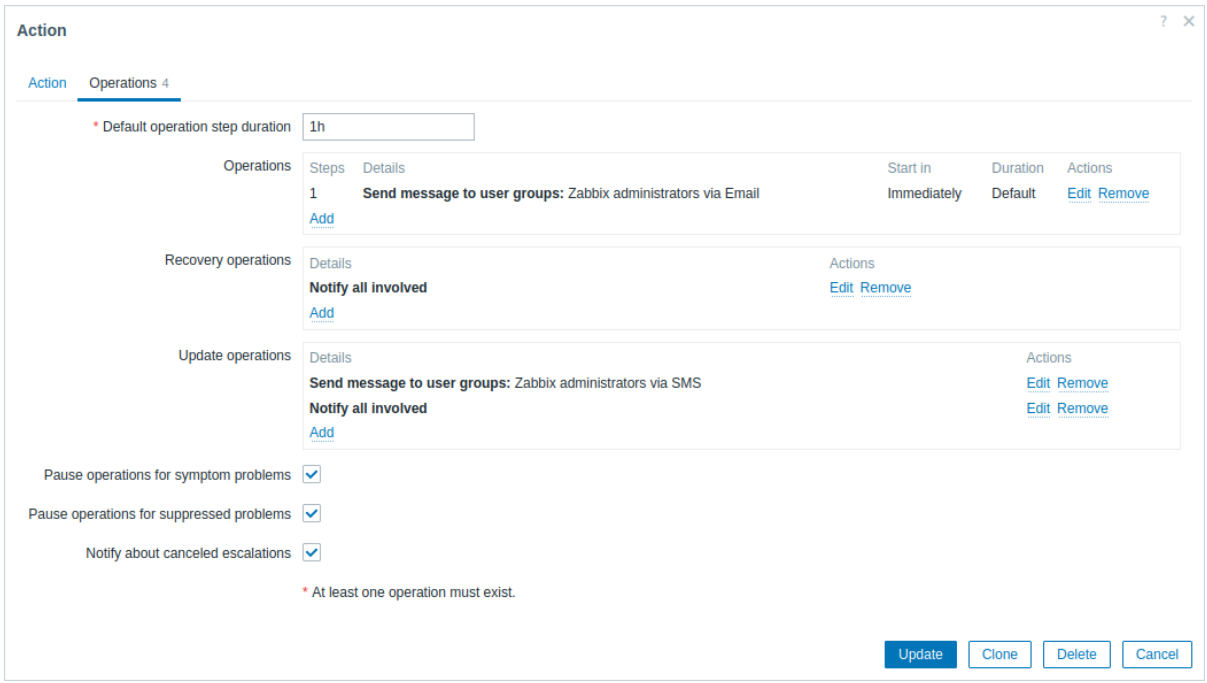
Dla zdarzeń wykrywania i autorejestracji dostępne są dodatkowe operacje:

- Dodaj host
- Usuń host
- Włącz host
- Wyłącz host
- Dodaj do grupy hostów
- Usuń z grupy hostów
- Dodaj tagi hosta
- Usuń tagi hosta
- Podłącz szablon
- Odłącz szablon
- Ustaw tryb inwentaryzacji hosta

Konfigurowanie operacji

Aby skonfigurować operację:

1. Przejdź do karty *Operations* w konfiguracji action.
2. Skonfiguruj szczegóły operacji.



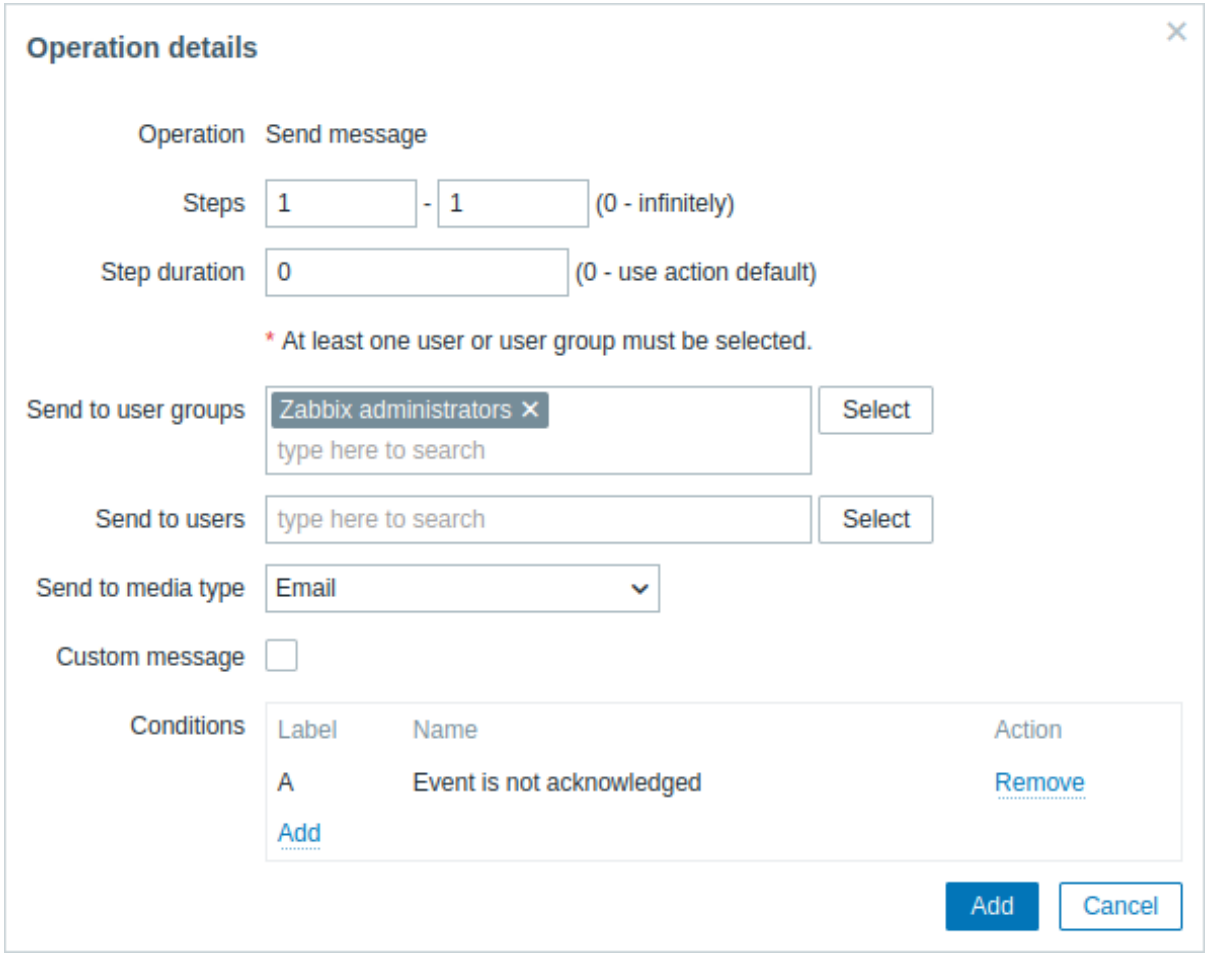
Ogólne atrybuty operacji:

Parameter	Description
Default operation step duration	Domyślny czas trwania jednego kroku operacji (od 60 sekund do 1 tygodnia). Na przykład godzinny czas trwania kroku oznacza, że jeśli operacja zostanie wykonana, przed przejściem do następnego kroku upłynie godzina. Obsługiwane są przyrostki czasu, np. 60s, 1m, 2h, 1d. Obsługiwane są makra użytkownika.
Operations	Wyświetlane są operacje akcji (jeśli istnieją) wraz z następującymi szczegółami: **Steps** - krok(i) eskalacji, do których przypisano operację. **Details** - typ operacji oraz jej odbiorca/element docelowy. Lista operacji wyświetla również używany typ nośnika (e-mail, SMS lub skrypt), a także imię i nazwisko (w nawiasach po nazwie użytkownika) odbiorcy powiadomienia. **Start in** - czas, po którym od wystąpienia zdarzenia operacja zostanie wykonana. **Duration (sec)** - wyświetlany jest czas trwania kroku. Jeśli krok korzysta z domyślnego czasu trwania, wyświetlana jest wartość *Default*, a jeśli używa niestandardowego czasu trwania, wyświetlany jest czas. **Actions** - wyświetlane są odnośniki umożliwiające edycję i usunięcie operacji.
Recovery operations	Wyświetlane są operacje akcji (jeśli istnieją) wraz z następującymi szczegółami: **Details** - typ operacji oraz jej odbiorca/element docelowy. Lista operacji wyświetla również używany typ nośnika (e-mail, SMS lub skrypt), a także imię i nazwisko (w nawiasach po nazwie użytkownika) odbiorcy powiadomienia. **Actions** - wyświetlane są odnośniki umożliwiające edycję i usunięcie operacji.
Update operations	Wyświetlane są operacje akcji (jeśli istnieją) wraz z następującymi szczegółami: **Details** - typ operacji oraz jej odbiorca/element docelowy. Lista operacji wyświetla również używany typ nośnika (e-mail, SMS lub skrypt), a także imię i nazwisko (w nawiasach po nazwie użytkownika) odbiorcy powiadomienia. **Actions** - wyświetlane są odnośniki umożliwiające edycję i usunięcie operacji.
Pause operations for symptom problems	Zaznacz to pole wyboru, aby wstrzymać operacje (po pierwszej operacji) w przypadku problemów typu symptom. Pamiętaj, że to ustawienie dotyczy tylko eskalacji problemów; operacje odzyskiwania i aktualizacji nie zostaną nim objęte. Ta opcja jest dostępna tylko dla *Trigger actions*.
Pause operations for suppressed problems	Zaznacz to pole wyboru, aby opóźnić rozpoczęcie operacji na czas trwania okresu konserwacji. Po zakończeniu konserwacji, gdy operacje zostaną rozpoczęte, wykonywane są wszystkie operacje, w tym te dotyczące zdarzeń, które wystąpiły podczas konserwacji. Pamiętaj, że to ustawienie dotyczy tylko eskalacji problemów; operacje odzyskiwania i aktualizacji nie zostaną nim objęte. Jeśli odznaczysz to pole wyboru, operacje będą wykonywane bez opóźnienia, nawet w okresie konserwacji. Ta opcja nie jest dostępna dla *Service actions*.
Notify about canceled escalations	Odznacz to pole wyboru, aby wyłączyć powiadomienia o anulowanych eskalacjach (gdy host, pozycja, wyzwalacz lub akcja jest wyłączona).

Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Aby skonfigurować szczegóły nowej operacji, kliknij Add w bloku *Operations*. Aby edytować istniejącą operację, kliknij Edit obok operacji. Otworzy się wyskakujące okno, w którym można edytować szczegóły kroku operacji.

Szczegóły operacji



Parametr	Opis
Operacja	Wybierz operację: **Wyślij wiadomość** - wysłanie wiadomości do użytkownika. **<nazwa polecenia zdalnego>** - wykonanie polecenia zdalnego. Polecenia są dostępne do wykonania, jeśli zostały wcześniej zdefiniowane w skryptach globalnych z wybranym zakresem *Operacja akcji*. Więcej operacji jest dostępnych dla zdarzeń związanych z wykrywaniem i automatyczną rejestracją (zobacz powyżej).
Kroki	Wybierz krok lub kroki, do których ma zostać przypisana operacja w harmonogramie eskalacji: **Od** - wykonanie rozpoczynające się od tego kroku. **Do** - wykonanie trwające do tego kroku (0=bez ograniczeń, wykonanie nie będzie ograniczone).
Czas tr-wa-nia kroku	Niestandardowy czas trwania tych kroków (0=użycie domyślnego czasu trwania kroku). Obsługiwane są przyrostki czasu, np. 60s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. Do tego samego kroku można przypisać kilka operacji. Jeśli dla tych operacji zdefiniowano różne czasy trwania kroku, uwzględniany i stosowany jest najkrótszy z nich.
Typ op-er-acji: wyślij wiado-mość	
Wyślij do grup użytkown-ików	Wybierz grupy użytkowników, do których ma zostać wysłana wiadomość. Grupa użytkowników musi mieć co najmniej uprawnienia „odczyt” uprawnienia do host, aby można było wysłać powiadomienie.

Parametr	Opis
Wyślij do użytkowników	Wybierz użytkowników, do których ma zostać wysłana wiadomość. Użytkownik musi mieć co najmniej uprawnienia „odczyt” uprawnienia do host, aby można było wysłać powiadomienie.
Wyślij do typu nośnika	Wyślij wiadomość do wszystkich dostępnych (skonfigurowanych i włączonych) typów nośników lub tylko do określonego typu.
Niestandardowa wiadomość	Jeśli wybrano tę opcję, można skonfigurować niestandardową wiadomość. W przypadku powiadomień o zdarzeniach wewnętrznych za pośrednictwem webhooków niestandardowa wiadomość jest wymagana.
Temat	Temat niestandardowej wiadomości. Temat może zawierać makra. Jest ograniczony do 255 znaków.
Wiadomość	Niestandardowa wiadomość. Wiadomość może zawierać makra. Jest ograniczona do określonej liczby znaków zależnie od typu bazy danych (więcej informacji zawiera sekcja Wysyłanie wiadomości).
Typ operacji: polecenie zdalne	
Lista celów	Wybierz cele, na których ma zostać wykonane polecenie: **Bieżący host** - polecenie jest wykonywane na hoście wyzwalacza, który spowodował zdarzenie problemu. Ta opcja nie będzie działać, jeśli wyzwalacz obejmuje wiele hostów. **Host** - wybierz host lub hosty, na których ma zostać wykonane polecenie. **Grupa hostów** - wybierz grupę lub grupy hostów, na których ma zostać wykonane polecenie. Określenie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. Polecenie zdalne zostanie więc wykonane również na hostach z zagnieżdżonych grup. Polecenie na hoście jest wykonywane tylko raz, nawet jeśli host pasuje więcej niż raz (np. z kilku grup hostów, indywidualnie i z grupy hostów). Lista celów nie ma znaczenia, jeśli niestandardowy skrypt jest wykonywany na serwerze Zabbix. Wybranie większej liczby celów w tym przypadku powoduje jedynie wielokrotne wykonanie skryptu na serwerze. Należy pamiętać, że w przypadku skryptów globalnych wybór celu zależy również od ustawienia *Grupa hostów* w konfiguracji skryptu globalnego. Opcja *Lista celów* nie jest dostępna dla *akcji usług*, ponieważ w tym przypadku polecenia zdalne są zawsze wykonywane na serwerze Zabbix.
Warunki	Warunek wykonania operacji: **Zdarzenie niepotwierdzone** - tylko gdy zdarzenie nie zostało potwierdzone. **Zdarzenie potwierdzone** - tylko gdy zdarzenie zostało potwierdzone. Opcja *Warunki* jest dostępna tylko dla *akcji wyzwalacza*.

Po zakończeniu kliknij *Dodaj*, aby dodać operację do listy *Operacje*.

Wysyłanie wiadomości

Przegląd

Wysłanie wiadomości jest jednym z najlepszych sposobów powiadamiania ludzi o problemie. Dlatego jest to jedna z podstawowych akcji oferowanych przez Zabbix.

Konfiguracja

Aby móc wysyłać i odbierać powiadomienia z Zabbix, należy:

- Zdefiniować media, do których będą wysyłane wiadomości.

Jeśli operacja ma miejsce poza okresem Aktywne zdefiniowanym dla wybranego medium w konfiguracji użytkownika, wiadomość nie zostanie wysłana.

Domyślny poziom ważności wyzwalacza („Niesklasyfikowany") **musi być** zaznaczony w konfiguracji mediów użytkownika, jeśli chcesz otrzymywać powiadomienia dotyczące zdarzeń innych niż wyzwalacze, takich jak wykrywanie, automatyczna rejestracja aktywnego agenta lub zdarzenia wewnętrzne.

- Skonfigurować operację działania, która wysyła wiadomość do jednego ze zdefiniowanych mediów.

Attention:
Zabbix wysyła powiadomienia tylko do tych użytkowników, którzy mają co najmniej uprawnienia „odczyt" do hosta, na którym wystąpiło zdarzenie. Co najmniej jeden host z wyrażenia wyzwalacza musi być dostępny.

Możesz skonfigurować niestandardowe scenariusze wysyłania wiadomości za pomocą eskalacji.

Aby prawidłowo odbierać i odczytywać wiadomości e-mail z Zabbix, serwery/klienci poczty e-mail muszą obsługiwać standardowy format „SMTP/MIME email", ponieważ Zabbix wysyła dane w kodowaniu UTF-8 (jeśli temat zawiera wyłącznie znaki ASCII, nie jest kodowany w UTF-8). Temat i treść wiadomości są kodowane w formacie base64 zgodnie ze standardem „SMTP/MIME email".

Limit długości wiadomości po rozwinięciu wszystkich makr jest taki sam jak limit długości wiadomości dla poleceń zdalnych.

Śledzenie wiadomości

Stan wiadomości wysłanych można wyświetlić w *Monitoring* > *Problemy*.

W kolumnie *Actions* można zobaczyć podsumowane informacje o wykonanych akcjach. Zielone liczby oznaczają wysłane wiadomości, czerwone - nieudane wiadomości. *In progress* oznacza, że akcja została zainicjowana. *Failed* informuje, że żadna akcja nie została wykonana pomyślnie.

Jeśli klikniesz czas zdarzenia, aby wyświetlić szczegóły zdarzenia, będziesz mógł zobaczyć szczegóły wiadomości wysłanych (lub niewysłanych) z powodu zdarzenia w bloku *Actions*.

W *Reports* > *Action log* zobaczysz szczegóły wszystkich wykonanych akcji dla tych zdarzeń, dla których skonfigurowano akcję.

Zdalne polecenia

Omówienie

Za pomocą zdalnych poleceń można zdefiniować, że określone, wstępnie zdefiniowane polecenie zostanie automatycznie wykonane na monitorowanym hoście po wystąpieniu określonego warunku.

Zdalne polecenia są więc zaawansowanym mechanizmem inteligentnego, proaktywnego monitorowania.

W najbardziej oczywistych zastosowaniach tej funkcji można:

- Automatycznie uruchomić ponownie aplikację (serwer WWW, middleware, CRM), jeśli nie odpowiada
- Użyć polecenia IPMI „reboot" do ponownego uruchomienia zdalnego serwera, jeśli nie odpowiada na żądania
- Automatycznie zwolnić miejsce na dysku (usuwając starsze pliki, czyszcząc katalog `/tmp`), jeśli zaczyna go brakować
- Migrować maszynę wirtualną z jednego serwera fizycznego na inny w zależności od obciążenia procesora
- Dodawać nowe węzły do środowiska chmurowego w przypadku niewystarczających zasobów procesora (dysku, pamięci lub innych)

Konfigurowanie akcji dla zdalnych poleceń jest podobne do konfigurowania akcji wysyłającej wiadomość; jedyna różnica polega na tym, że Zabbix wykonuje polecenie zamiast wysyłać wiadomość.

Zdalne polecenia mogą być wykonywane przez serwer Zabbix, proxy Zabbix lub agenta Zabbix. Należy pamiętać, że:

- Zdalne polecenia na agencie Zabbix mogą być wykonywane bezpośrednio przez serwer Zabbix lub za pośrednictwem proxy Zabbix.
- Zdalne polecenia wykonywane przez serwer Zabbix są uruchamiane zgodnie z opisem w sekcji Wykonywanie poleceń, w tym z kontrolą kodu wyjścia.
- Zdalne polecenia są wykonywane nawet wtedy, gdy docelowy host znajduje się w trybie konserwacji.

Zdalne polecenia są domyślnie wyłączone zarówno na agencie Zabbix, jak i na proxy Zabbix. Można je włączyć w następujący sposób:

- W przypadku zdalnych poleceń wykonywanych przez agenta Zabbix dodaj parametr `AllowKey=system.run[<command>,*]` lub `AllowKeyRegexp=^system\.run\[<command>,.*\]$` dla każdego dozwolonego polecenia w konfiguracji agenta, gdzie `*` odpowiada dowolnemu trybowi wykonywania (`wait` lub `nowait`).

- W przypadku zdalnych poleceń wykonywanych przez proxy Zabbix ustaw parametr `EnableRemoteCommands` na 1 w konfiguracji proxy.

Należy pamiętać, że nie trzeba włączać zdalnych poleceń na proxy Zabbix, jeśli są one wykonywane przez agenta Zabbix na hoście monitorowanym przez proxy.

Attention:
Nie używaj `AllowKey=system.run[*]` ani `AllowKeyRegexp=^system\.run\[.*\]$`, chyba że jest to konieczne, ponieważ umożliwiają one wykonywanie dowolnych poleceń na hoście. Ze względów bezpieczeństwa zaleca się zezwalanie wyłącznie na dokładnie określone polecenia, natomiast `system.run[*]` jest domyślnie blokowane (`DenyKey=system.run[*]`).

Zobacz także Ograniczanie kontroli agenta.

Limit zdalnych poleceń

Limit zdalnego polecenia po rozwinięciu wszystkich makr zależy od typu bazy danych i zestawu znaków (znaki spoza ASCII wymagają do zapisania więcej niż jednego bajtu):

Database	Limit in characters	Limit in bytes
MySQL	65535	65535
PostgreSQL	65535	bez limitu
SQLite (tylko Zabbix proxy)	65535	bez limitu

Dane wyjściowe wykonania zdalnego polecenia (wartość zwracana) są ograniczone do 16 MB (włącznie z końcowymi białymi znakami, które są obcinane). Limit zdalnego polecenia IPMI zależy od zainstalowanej biblioteki IPMI. Należy pamiętać, że limity bazy danych mają zastosowanie do wszystkich zdalnych poleceń.

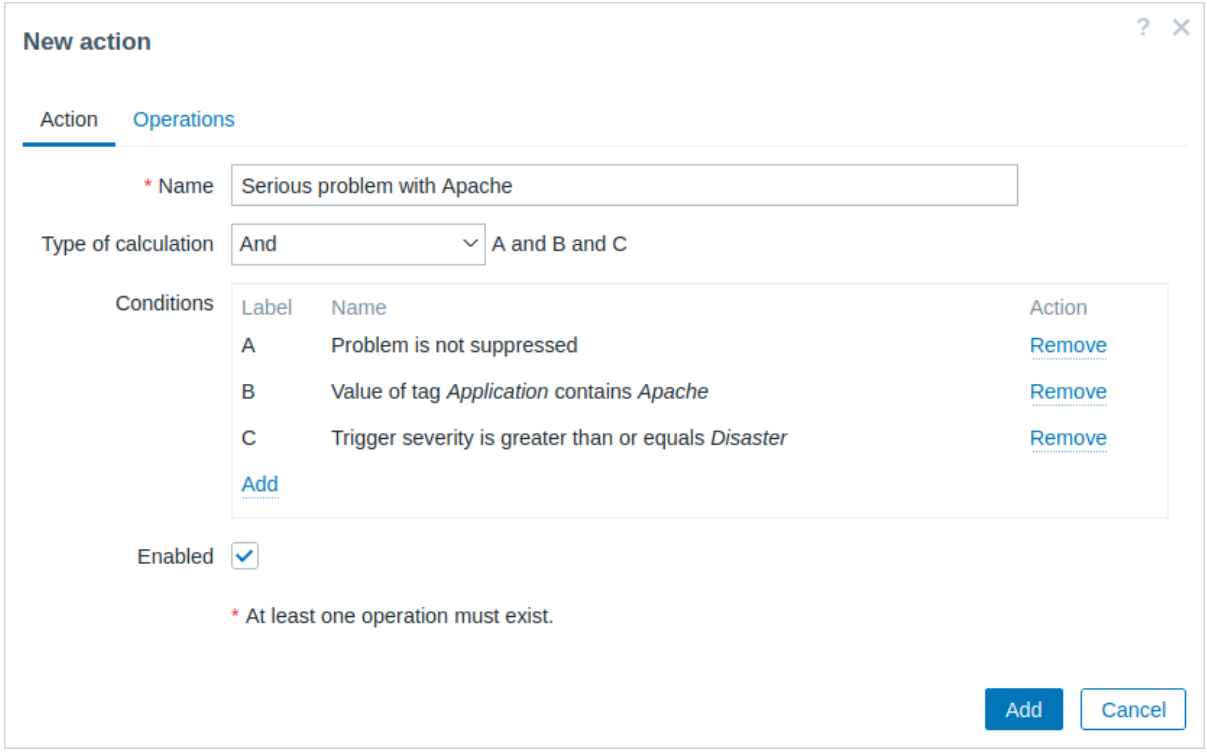
Konfiguracja

Zdalne polecenia wykonywane przez agenta Zabbix muszą najpierw zostać dozwolone w konfiguracji agenta za pomocą parametru `AllowKey` lub `AllowKeyRegexp`.

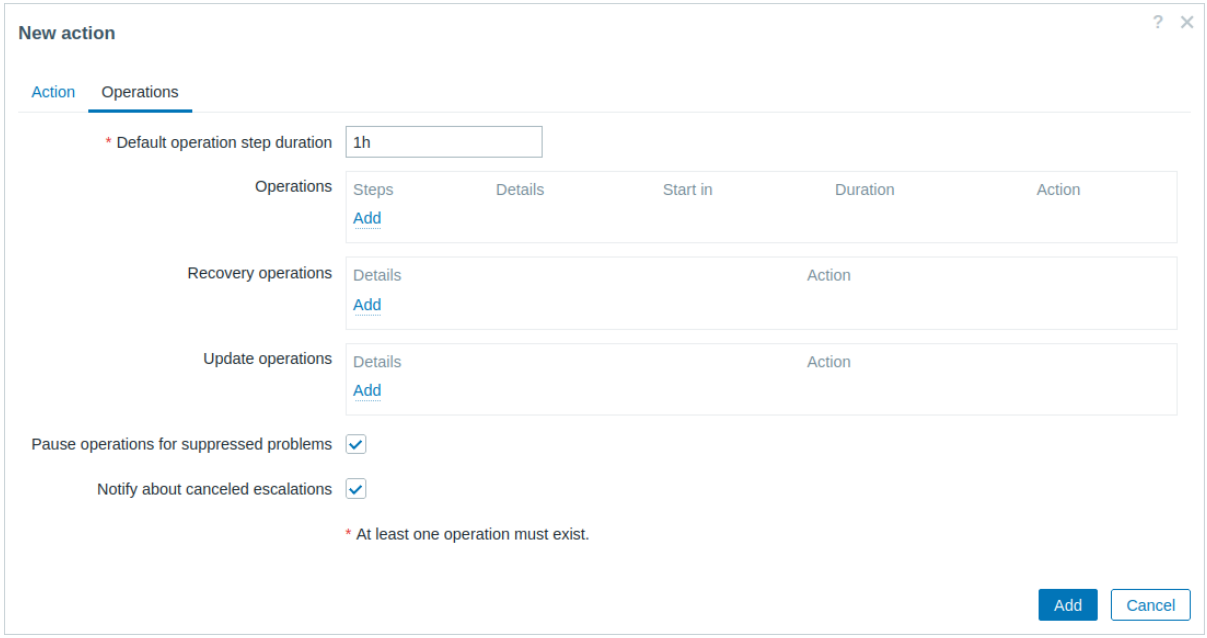
Upewnij się, że parametr `AllowKey=system.run[<command>,*]` lub `AllowKeyRegexp=^system\.run\[<command>,.*\]$` został dodany do konfiguracji agenta dla każdego dozwolonego polecenia, aby umożliwić jego wykonanie w trybie `nowait`. Po zmianie parametru uruchom ponownie demona agenta.

Następnie podczas konfigurowania nowej akcji w sekcji *Alerts* > *Actions* > *Trigger actions*:

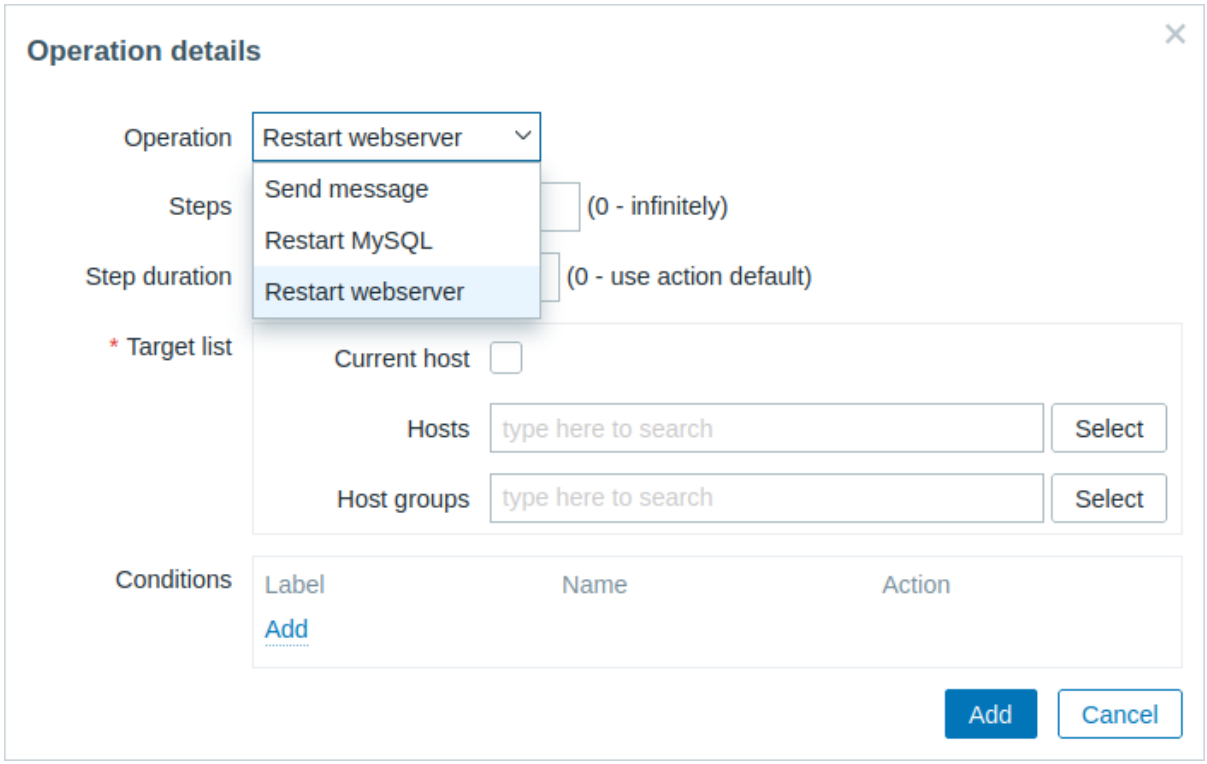
1. Zdefiniuj odpowiednie warunki, na przykład ustaw, aby akcja była aktywowana w przypadku dowolnych problemów o poziomie katastrofalnym dotyczących jednej z aplikacji Apache.



2. Na karcie *Operations* kliknij *Add* w sekcji *Operations*, *Recovery operations* lub *Update operations*.



3. Wybierz jeden ze wstępnie zdefiniowanych skryptów z listy rozwijanej *Operation* i ustaw dla skryptu *Target list*.



Predefiniowane skrypty

Skrypty dostępne dla operacji akcji (webhook, skrypt, SSH, Telnet, IPMI) są definiowane w globalnych skryptach z wybranym zakresem *Action operation*.

Na przykład:

```
sudo /etc/init.d/apache restart
```

W tym przypadku Zabbix spróbuje zrestartować proces Apache. W przypadku tego polecenia upewnij się, że jest ono wykonywane na agent Zabbix (kliknij przycisk *Zabbix agent* przy opcji *Execute on*).

Attention:
Zwróć uwagę na użycie sudo - użytkownik Zabbix nie ma domyślnie uprawnień do restartowania usług systemowych. Poniżej znajdziesz wskazówki, jak skonfigurować sudo.

Note:
Począwszy od Zabbix agent 7.0, polecenia zdalne mogą być również wykonywane na agencie działającym w trybie aktywnym. Agent Zabbix - niezależnie od tego, czy działa w trybie aktywnym, czy pasywnym - powinien działać na zdalnym host, i wykonuje polecenia w tle.

Polecenia zdalne na agent Zabbix są wykonywane bez limitu czasu przez klucz `system.run[,nowait]` i nie są sprawdzane pod kątem wyników wykonania. Na serwer Zabbix i proxy Zabbix polecenia zdalne są wykonywane z limitem czasu ustawionym w parametrze `TrapperTimeout` pliku `zabbix_server.conf` lub `zabbix_proxy.conf` i są sprawdzane pod kątem wyników wykonania. Dodatkowe informacje znajdziesz w sekcji *Script timeout*.

Uprawnienia dostępu

Upewnij się, że użytkownik `zabbix` ma uprawnienia do wykonywania skonfigurowanych poleceń. Można rozważyć użycie `sudo`, aby nadać dostęp do poleceń wymagających uprawnień. Aby skonfigurować dostęp, wykonaj jako root:

```
visudo
```

Przykładowe wiersze, które można umieścić w pliku `sudoers`:

```
# pozwala użytkownikowi 'zabbix' uruchamiać wszystkie polecenia bez hasła.
zabbix ALL=NOPASSWD: ALL

# pozwala użytkownikowi 'zabbix' restartować apache bez hasła.
zabbix ALL=NOPASSWD: /etc/init.d/apache restart
```

Note:
W niektórych systemach plik `sudoers` uniemożliwi użytkownikom spoza lokalnego systemu wykonywanie poleceń. Aby to zmienić, zakomentuj opcję `requiretty` w `/etc/sudoers`.

Zdalne polecenia przy wielu interfejsach

Jeśli system docelowy ma wiele interfejsów wybranego typu (agent Zabbix lub IPMI), zdalne polecenia będą wykonywane na domyślnym interfejsie.

Możliwe jest wykonywanie zdalnych poleceń przez SSH i Telnet przy użyciu interfejsu innego niż interfejs agenta Zabbix. Dostępny interfejs do użycia jest wybierany w następującej kolejności:

- Domyślny interfejs agenta Zabbix
- Domyślny interfejs SNMP
- Domyślny interfejs JMX
- Domyślny interfejs IPMI

Zdalne polecenia IPMI

Dla zdalnych poleceń IPMI należy używać następującej składni:

```
<command> [<value>]
```

gdzie

- <command> - jedno z poleceń IPMI bez spacji
- <value> - 'on', 'off' lub dowolna nieujemna liczba całkowita. <value> jest parametrem opcjonalnym.

Przykłady

Przykłady globalnych skryptów, które mogą być używane jako zdalne polecenia w operacjach akcji.

Przykład 1

Ponowne uruchomienie systemu Windows w określonych warunkach.

Aby automatycznie uruchomić ponownie system Windows po wykryciu problemu przez Zabbix, zdefiniuj następujący skrypt:

Parametr skryptu	Wartość
Zakres	'Action operation'
Typ	'Script'
Polecenie	`c:\windows\system32\shutdown.exe -r -f`

Przykład 2

Uruchom ponownie host przy użyciu sterowania IPMI.

Parametr skryptu	Wartość
Zakres	'Operacja akcji'
Typ	'IPMI'
Polecenie	reset

Przykład 3

Wyłącz zasilanie hosta za pomocą sterowania IPMI.

Parametr skryptu	Wartość
Zakres	'Operacja akcji'
Typ	'IPMI'
Polecenie	power off

Dodatkowe operacje

Przegląd

W tej sekcji możesz znaleźć szczegóły dodatkowych operacji dla zdarzeń wykrywania/autorejestracji.

Dodawanie hosta

Hosty są dodawane podczas procesu wykrywania, natychmiast po wykryciu hosta, a nie dopiero na końcu procesu wykrywania.

Note:
Ponieważ wykrywanie sieci może zająć trochę czasu z powodu wielu niedostępnych hostów/usług, zaleca się cierpliwość i używanie rozsądnych zakresów IP.

Podczas dodawania hosta jego nazwa jest ustalana przez standardową funkcję **gethostbyname**. Jeśli host może zostać rozwiązany, używana jest rozwiązana nazwa. Jeśli nie, używany jest adres IP. Ponadto, jeśli jako nazwa hosta musi zostać użyty adres IPv6, wszystkie ":" (dwukropki) są zastępowane przez "_" (podkreślenia), ponieważ dwukropki nie są dozwolone w nazwach hostów.

Attention:
Jeśli wykrywanie jest wykonywane przez proxy, obecnie wyszukiwanie nazwy hosta nadal odbywa się na serwerze Zabbix.

Attention:
Jeśli host o tej samej nazwie co nowo wykryty już istnieje w konfiguracji Zabbix, Zabbix doda **_N** do nazwy hosta, gdzie **N** jest rosnącą liczbą, zaczynając od 2.

Używanie makr w wiadomościach

Przegląd

W tematach wiadomości i treści wiadomości można używać makr w celu bardziej efektywnego raportowania problemów.

Oprócz wielu wbudowanych makr obsługiwane są również makra użytkownika oraz makra wyrażeń. Dostępna jest pełna lista makr obsługiwanych przez Zabbix.

Przykłady

Przykłady tutaj ilustrują, jak można używać makr w wiadomościach.

Przykład 1

Temat wiadomości:

```
Problem: {TRIGGER.NAME}
```

Po otrzymaniu wiadomości temat wiadomości zostanie zastąpiony czymś takim jak:

```
Problem: Obciążenie procesora jest zbyt wysokie na serwerze Zabbix
```

Przykład 2

Wiadomość:

```
Obciążenie procesora wynosi: {?last(/zabbix.zabbix.com/system.cpu.load[,avg1])}
```

Po otrzymaniu wiadomości treść wiadomości zostanie zastąpiona czymś takim jak:

```
Obciążenie procesora wynosi: 1.45
```

Przykład 3

Wiadomość:

```
Ostatnia wartość: {?last(/{HOST.HOST}/{ITEM.KEY})}
MAX z 15 minut: {?max(/{HOST.HOST}/{ITEM.KEY},15m)}
MIN z 15 minut: {?min(/{HOST.HOST}/{ITEM.KEY},15m)}
```

Po otrzymaniu wiadomości jej treść zostanie zastąpiona czymś takim jak:

```
Ostatnia wartość: 1.45
MAX z 15 minut: 2.33
MIN z 15 minut: 1.01
```

Przykład 4

Wiadomość:

```
http://<server_ip_or_name>/zabbix/tr_events.php?triggerid={TRIGGER.ID}&eventid={EVENT.ID}
```

Po otrzymaniu wiadomości będzie ona zawierać link do strony *Szczegóły zdarzenia*, która zawiera informacje o zdarzeniu, jego wyzwalaczu oraz listę najnowszych zdarzeń wygenerowanych przez ten sam wyzwalacz.

Przykład 5

Informowanie o wartościach z kilku hostów w wyrażeniu wyzwalacza.

Wiadomość:

```
Nazwa problemu: {TRIGGER.NAME}
Wyrażenie wyzwalacza: {TRIGGER.EXPRESSION}

1. Wartość pozycji na hoście {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1})
2. Wartość pozycji na hoście {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})
```

Po otrzymaniu wiadomości zostanie ona zastąpiona czymś takim:

```
Nazwa problemu: Obciążenie procesora jest zbyt wysokie na lokalnym hoście
Wyrażenie wyzwalacza: last(/Myhost/system.cpu.load[percpu,avg1])>5 or last(/Myotherhost/system.cpu.load[pe

1. Wartość pozycji na hoście Myhost: 0.83 (Obciążenie procesora (średnia 1-minutowa na rdzeń))
2. Wartość pozycji na hoście Myotherhost: 5.125 (Obciążenie procesora (średnia 1-minutowa na rdzeń))
```

Przykład 6

Otrzymywanie szczegółów zarówno zdarzenia problemu, jak i zdarzenia odzyskania w wiadomości odzyskania:

Wiadomość:

```
Problem:

ID zdarzenia: {EVENT.ID}
Wartość zdarzenia: {EVENT.VALUE}
Status zdarzenia: {EVENT.STATUS}
Czas zdarzenia: {EVENT.TIME}
Data zdarzenia: {EVENT.DATE}
Wiek zdarzenia: {EVENT.AGE}
Potwierdzenie zdarzenia: {EVENT.ACK.STATUS}
Historia aktualizacji zdarzenia: {EVENT.UPDATE.HISTORY}

Odzyskanie:
```

```
ID zdarzenia: {EVENT.RECOVERY.ID}
Wartość zdarzenia: {EVENT.RECOVERY.VALUE}
Status zdarzenia: {EVENT.RECOVERY.STATUS}
Czas zdarzenia: {EVENT.RECOVERY.TIME}
Data zdarzenia: {EVENT.RECOVERY.DATE}
Dane operacyjne: {EVENT.OPDATA}
```

Po otrzymaniu wiadomości makra zostaną zastąpione czymś takim:

```
Problem:

ID zdarzenia: 21874
Wartość zdarzenia: 1
Status zdarzenia: PROBLEM
Czas zdarzenia: 13:04:30
Data zdarzenia: 2018.01.02
Wiek zdarzenia: 5m 0s
Potwierdzenie zdarzenia: Tak
Historia aktualizacji zdarzenia: 2018.01.02 13:05:51 "John Smith (Admin)"
Działania: potwierdzono.

Odzyskanie:

ID zdarzenia: 21896
Wartość zdarzenia: 0
Status zdarzenia: OK
Czas zdarzenia: 13:10:07
Data zdarzenia: 2018.01.02
Dane operacyjne: Bieżąca wartość to 0.83
```

Operacje odzyskiwania

Przegląd

Operacje odzyskiwania umożliwiają otrzymywanie powiadomień, gdy problemy zostaną rozwiązane.

Zarówno wiadomości, jak i zdalne polecenia są obsługiwane w operacjach odzyskiwania. Chociaż można dodać kilka operacji, eskalacja nie jest obsługiwana — wszystkie operacje są przypisane do jednego kroku i dlatego zostaną wykonane jednocześnie.

Przypadki użycia

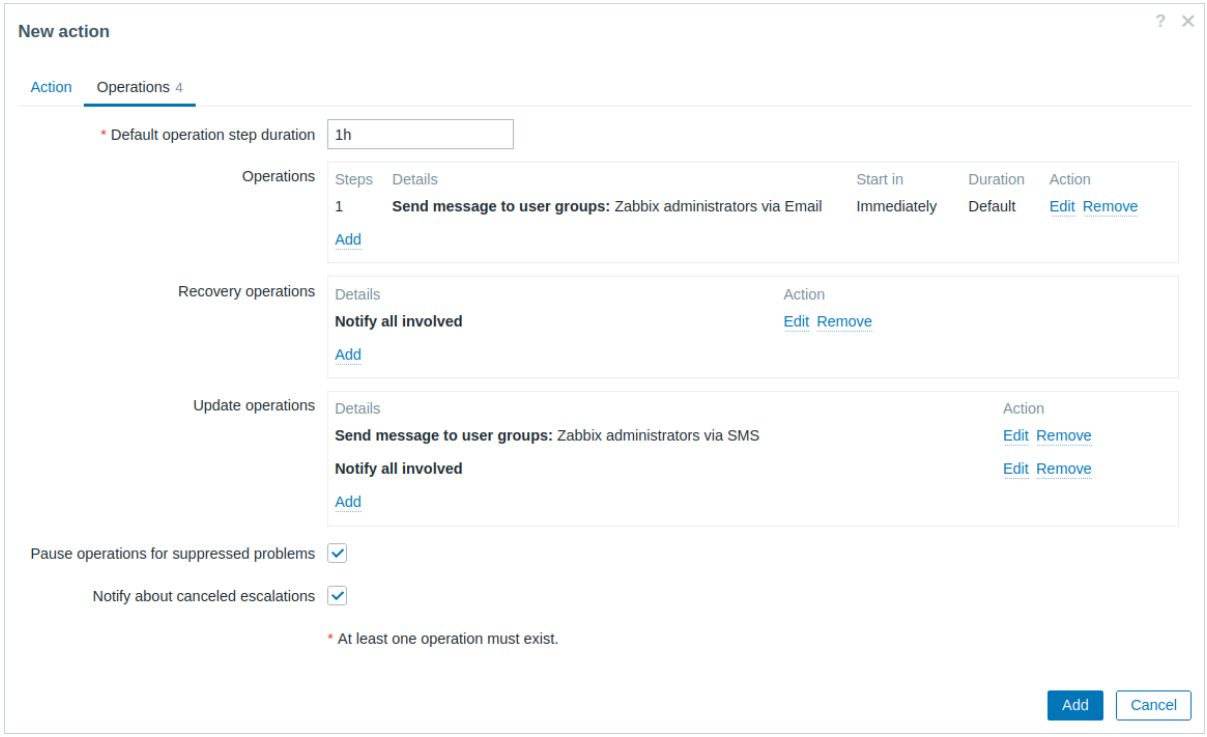
Niektóre przypadki użycia operacji odzyskiwania są następujące:

1. Powiadom przy odzyskaniu wszystkich użytkowników, którzy zostali powiadomieni o problemie:
   - Wybierz *Powiadom wszystkich zaangażowanych* jako typ operacji.
2. Miej wiele operacji przy odzyskaniu: wyślij powiadomienie i wykonaj zdalne polecenie:
   - Dodaj typy operacji do wysyłania wiadomości i wykonywania polecenia.
3. Otwórz zgłoszenie w zewnętrznym systemie helpdesk/ticketing i zamknij je po rozwiązaniu problemu:
   - Utwórz zewnętrzny skrypt, który komunikuje się z systemem helpdesk.
   - Utwórz akcję z operacją, która wykonuje ten skrypt i w ten sposób otwiera zgłoszenie.
   - Skonfiguruj operację odzyskiwania, która wykonuje ten skrypt z innymi parametrami i zamyka zgłoszenie.
   - Użyj makra {EVENT.ID}, aby odwołać się do pierwotnego problemu.

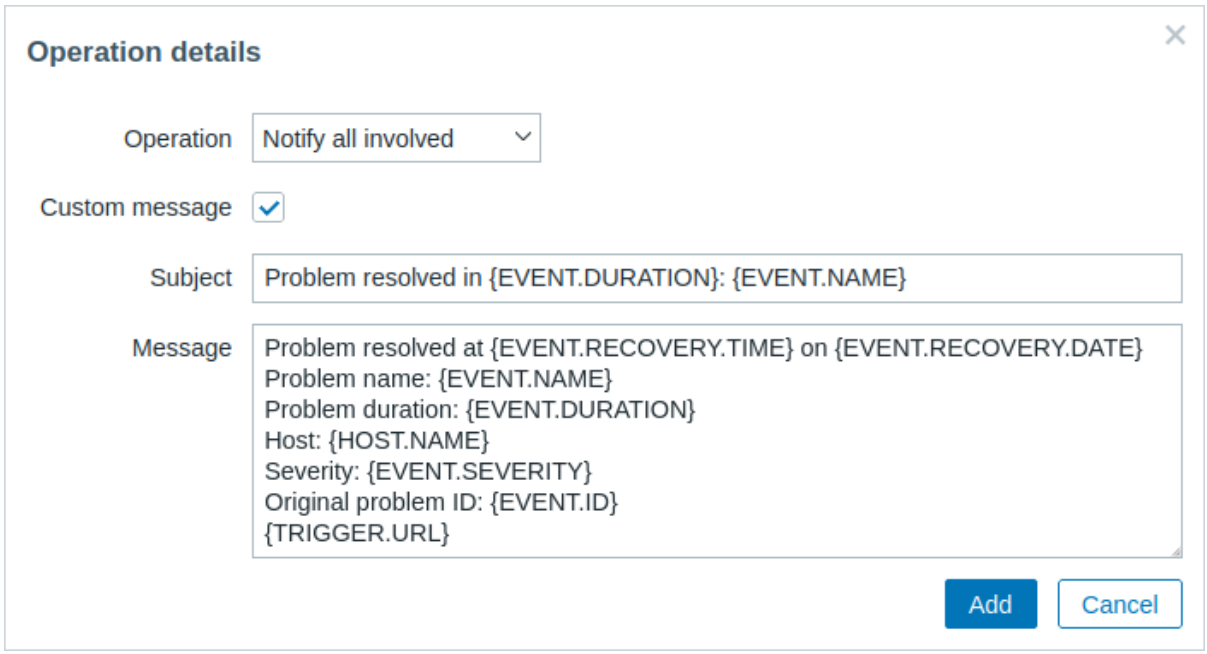
Konfigurowanie operacji odzyskiwania

Aby skonfigurować operację odzyskiwania, wykonaj następujące kroki:

1. Przejdź do zakładki *Operations* w formularzu konfiguracji akcji.
2. Aby skonfigurować nową operację odzyskiwania, kliknij *Add* w sekcji *Recovery operations*. Aby edytować istniejącą operację, kliknij *Edit* obok operacji.
3. Skonfiguruj szczegóły operacji.



Szczegóły operacji odzyskiwania



Dla zdarzeń odzyskiwania dostępne są trzy typy operacji:

- **Wyślij wiadomość** - wyślij wiadomość odzyskiwania do określonego użytkownika;
- **Powiadom wszystkich zaangażowanych** - wyślij wiadomość odzyskiwania do wszystkich użytkowników, którzy zostali powiadomieni o zdarzeniu problemowym;
- **<nazwa zdalnego polecenia>** - wykonaj zdalne polecenie. Polecenia są dostępne do wykonania, jeśli zostały wcześniej zdefiniowane w skryptach globalnych z wybranym zakresem *Action operation*.

Parametry dla każdego typu operacji opisano poniżej. Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką. Po zakończeniu kliknij *Dodaj*, aby dodać operację do listy *Operacje odzyskiwania*.

Note:
Zwróć uwagę, że jeśli ten sam odbiorca jest zdefiniowany w kilku typach operacji bez określonej opcji *Własna wiadomość*, zduplikowane powiadomienia nie są wysyłane.

Typ operacji: wyślij wiadomość

Parametr	Opis
Wyślij do grup użytkowników	Wybierz grupy użytkowników, do których ma zostać wysłana wiadomość o odzyskaniu. Aby otrzymać powiadomienie, grupa użytkowników musi mieć co najmniej uprawnienia „odczyt” do hosta permissions.
Wyślij do użytkowników	Wybierz użytkowników, do których ma zostać wysłana wiadomość o odzyskaniu. Aby otrzymać powiadomienie, użytkownik musi mieć co najmniej uprawnienia „odczyt” do hosta permissions.
Wyślij do typu nośnika	Wyślij domyślną wiadomość o odzyskaniu do wszystkich dostępnych (skonfigurowanych i włączonych) typów nośników lub tylko do jednego określonego typu.
Własna wiadomość	Jeśli ta opcja jest zaznaczona, można zdefiniować własną wiadomość.
Temat	Temat własnej wiadomości. Temat może zawierać makra.
Wiadomość	Własna wiadomość. Wiadomość może zawierać makra.

Typ operacji: polecenie zdalne

Parametr	Opis
Lista celów	Wybierz cele, na których ma zostać wykonane polecenie: **Bieżący host** - polecenie jest wykonywane na hoście wyzwalacza, który spowodował zdarzenie problemu. Ta opcja nie będzie działać, jeśli wyzwalacz obejmuje wiele hostów. **Host** - wybierz host(y), na których ma zostać wykonane polecenie. **Grupa hostów** - wybierz grupę(y) hostów, na których ma zostać wykonane polecenie. Określenie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. W związku z tym polecenie zdalne zostanie również wykonane na hostach z zagnieżdżonych grup. Polecenie na hoście jest wykonywane tylko raz, nawet jeśli host pasuje więcej niż raz (np. z kilku grup hostów; indywidualnie i z grupy hostów). Lista celów nie ma znaczenia, jeśli polecenie jest wykonywane na serwerze Zabbix. Wybranie większej liczby celów w tym przypadku powoduje jedynie wielokrotne wykonanie polecenia na serwerze. Należy pamiętać, że w przypadku skryptów globalnych wybór celu zależy również od ustawienia *Grupa hostów* w konfiguracji skryptu globalnego.

Typ operacji: powiadom wszystkich zaangażowanych

Parametr		Opis
Własna wiadomość		Jeśli zaznaczone, można zdefiniować własną wiadomość.
	Temat	Temat własnej wiadomości. Temat może zawierać makra.
	Wiadomość	Własna wiadomość. Wiadomość może zawierać makra.

Operacje aktualizacji

Przegląd

Operacje aktualizacji są dostępne w akcjach dla następujących źródeł zdarzeń:

- *Wyzwalacze* - gdy problemy są aktualizowane przez innych użytkowników, tzn. komentowane, potwierdzane, gdy zmieniono ich ważność, zamykane (ręcznie);
- *Usługi* - gdy ważność usługi uległa zmianie, ale usługa nadal nie została przywrócona.

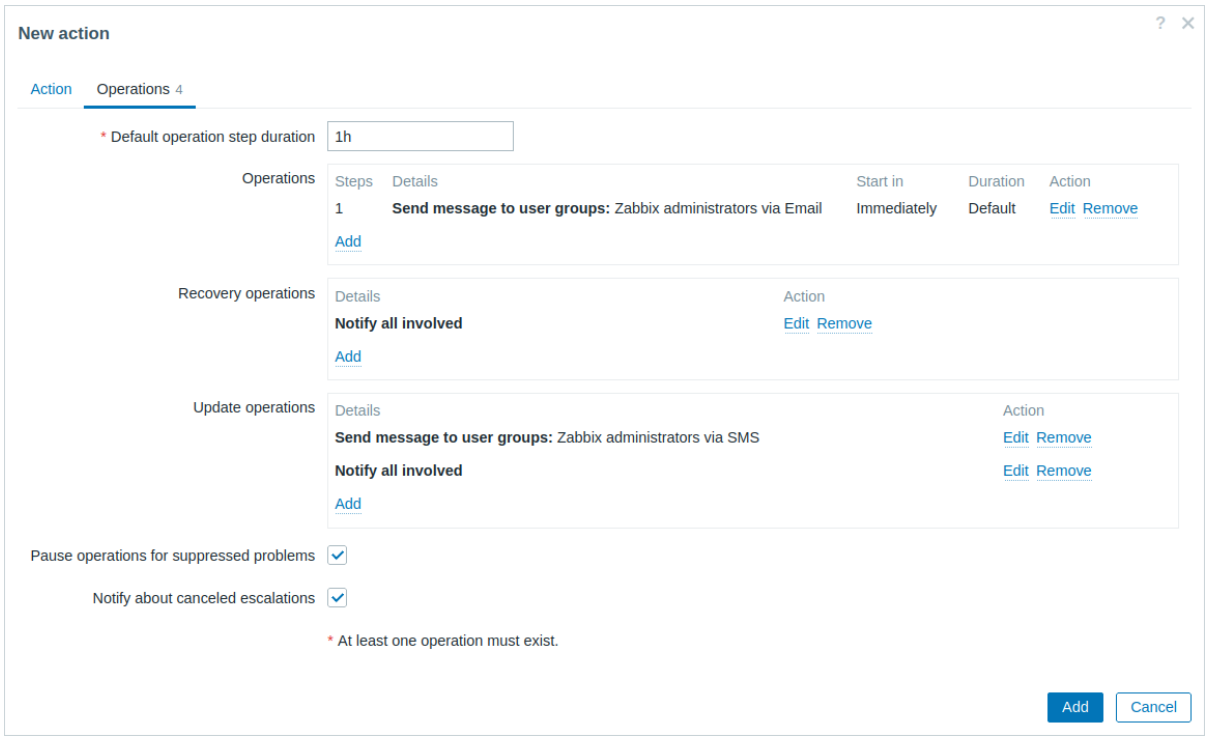
Należy pamiętać, że użytkownicy nie otrzymują powiadomień o własnych aktualizacjach.

W operacjach aktualizacji obsługiwane są zarówno wiadomości, jak i zdalne polecenia. Chociaż można dodać kilka operacji, eskalacja nie jest obsługiwana - wszystkie operacje są przypisane do jednego kroku, a zatem zostaną wykonane jednocześnie.

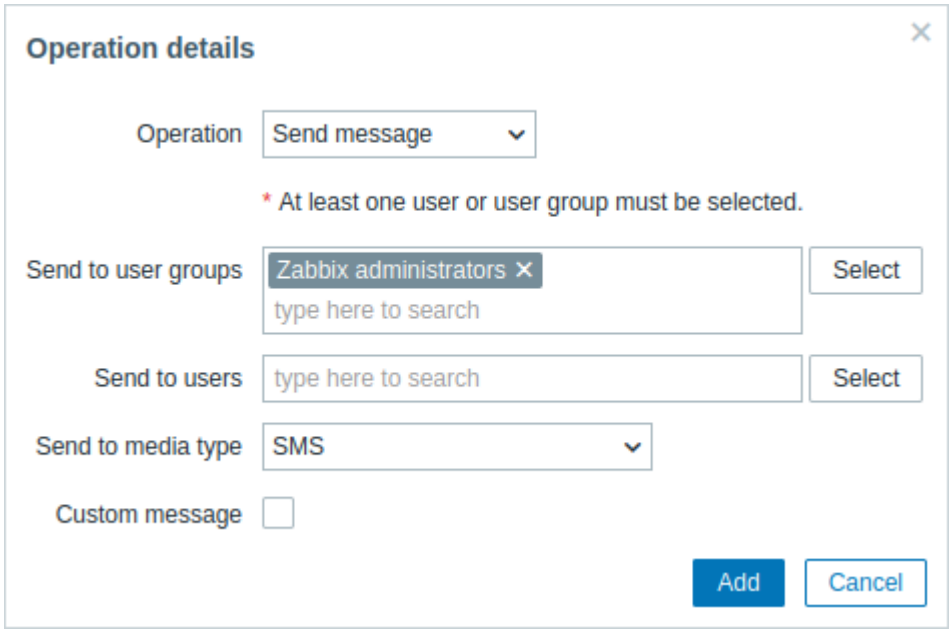
Konfigurowanie operacji aktualizacji

Aby skonfigurować operację aktualizacji, wykonaj następujące kroki:

1. Przejdź do zakładki *Operacje* w formularzu konfiguracji akcji.
2. Aby skonfigurować nową operację aktualizacji, kliknij *Dodaj* w sekcji *Operacje aktualizacji*. Aby edytować istniejącą operację, kliknij *Edytuj* obok operacji.
3. Skonfiguruj szczegóły operacji.



Szczegóły operacji aktualizacji



Operacje aktualizacji oferują ten sam zestaw parametrów co Operacje odzyskiwania.

Eskalacje

Przegląd

Za pomocą eskalacji można tworzyć niestandardowe scenariusze wysyłania powiadomień lub wykonywania zdalnych poleceń.

W praktyce oznacza to, że:

- Użytkownicy mogą być natychmiast informowani o nowych problemach.
- Powiadomienia mogą być powtarzane do momentu rozwiązania problemu.
- Wysłanie powiadomienia może zostać opóźnione.
- Powiadomienia mogą być eskalowane do innej, „wyższej” grupy użytkowników.
- Zdalne polecenia mogą być wykonywane natychmiast lub wtedy, gdy problem nie zostanie rozwiązany przez dłuższy czas.

Akcje są eskalowane na podstawie **kroku eskalacji**. Każdy krok ma określony czas trwania.

Można zdefiniować zarówno domyślny czas trwania, jak i niestandardowy czas trwania poszczególnego kroku. Minimalny czas trwania jednego kroku eskalacji wynosi 60 sekund.

Można uruchamiać akcje, takie jak wysyłanie powiadomień lub wykonywanie poleceń, od dowolnego kroku. Krok pierwszy służy do działań natychmiastowych. Jeśli chcesz opóźnić akcję, możesz przypisać ją do późniejszego kroku. Dla każdego kroku można zdefiniować kilka akcji.

Liczba kroków eskalacji nie jest ograniczona.

Eskalacje są definiowane podczas konfigurowania operacji. Eskalacje są obsługiwane tylko dla operacji problemowych, a nie dla odzyskiwania.

Różne aspekty zachowania eskalacji

Rozważmy, co dzieje się w różnych sytuacjach, jeśli akcja zawiera kilka kroków eskalacji.

Sytuacja	Zachowanie
Odpowiedni host przechodzi w tryb konserwacji po wysłaniu powiadomienia o początkowym problemie	W zależności od ustawienia *Wstrzymuj operacje dla wyciszonych problemów* w konfiguracji akcji wszystkie pozostałe kroki eskalacji są wykonywane albo z opóźnieniem spowodowanym okresem konserwacji, albo bez opóźnienia. Okres konserwacji nie anuluje operacji.
*Okres zdefiniowany w warunku akcji **Okres** kończy się po wysłaniu początkowego powiadomienia*	Wszystkie pozostałe kroki eskalacji są wykonywane. Warunek *Okres* nie może zatrzymać operacji; ma wpływ na to, kiedy akcje są uruchamiane lub nie są uruchamiane, a nie na operacje.
Problem rozpoczyna się podczas konserwacji i trwa (nie zostaje rozwiązany) po zakończeniu konserwacji	W zależności od ustawienia *Wstrzymuj operacje dla wyciszonych problemów* w konfiguracji akcji wszystkie kroki eskalacji są wykonywane albo od momentu zakończenia konserwacji, albo natychmiast.
Problem rozpoczyna się podczas konserwacji bez danych i trwa (nie zostaje rozwiązany) po zakończeniu konserwacji	Należy poczekać na uruchomienie wyzwalacza, zanim zostaną wykonane wszystkie kroki eskalacji.
Różne eskalacje następują w krótkich odstępach czasu i nakładają się na siebie	Wykonanie każdej nowej eskalacji zastępuje poprzednią eskalację, jednak co najmniej jeden krok eskalacji jest zawsze wykonywany w ramach poprzedniej eskalacji. To zachowanie ma znaczenie w przypadku akcji dotyczących zdarzeń tworzonych przy KAŻDEJ ocenie problemu przez wyzwalacz.
Podczas trwającej eskalacji (na przykład podczas wysyłania wiadomości), na podstawie dowolnego typu zdarzenia: - akcja zostaje wyłączona Na podstawie zdarzenia wyzwalacza: - wyzwalacz zostaje wyłączony - host lub pozycja zostaje wyłączona Na podstawie wewnętrznego zdarzenia dotyczącego wyzwalaczy: - wyzwalacz zostaje wyłączony Na podstawie wewnętrznego zdarzenia dotyczącego pozycji/reguł wykrywania niskiego poziomu: - pozycja zostaje wyłączona - host zostaje wyłączony	Trwająca wiadomość zostaje wysłana, a następnie wysyłana jest jeszcze jedna wiadomość dotycząca eskalacji. Wiadomość uzupełniająca będzie zawierać na początku treści komunikat o anulowaniu (*UWAGA: Eskalacja anulowana*) wraz z podaniem przyczyny (na przykład *UWAGA: Eskalacja anulowana: akcja „<Nazwa akcji>” wyłączona*). Dzięki temu odbiorca zostaje poinformowany, że eskalacja została anulowana i nie zostaną wykonane żadne kolejne kroki. Ta wiadomość jest wysyłana do wszystkich odbiorców, którzy wcześniej otrzymali powiadomienia. Przyczyna anulowania jest również zapisywana w pliku dziennika serwera (od poziomu debugowania 3=Warning). Należy pamiętać, że wiadomość *Eskalacja anulowana* jest wysyłana również wtedy, gdy operacje zostały zakończone, ale skonfigurowano operacje przywrócenia i nie zostały one jeszcze wykonane.

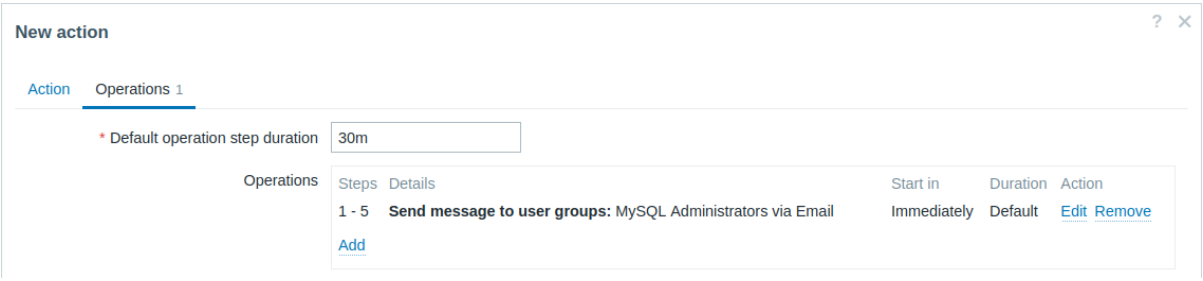
Sytuacja	Zachowanie
Podczas trwającej eskalacji (na przykład podczas wysyłania wiadomości) akcja zostaje usunięta	Nie są wysyłane żadne kolejne wiadomości. Informacja jest zapisywana w pliku dziennika serwera (od poziomu debugowania 3=Warning), na przykład: `escalation canceled: action id:334 deleted`

Przykłady eskalacji

Przykład 1

Wysyłanie powtarzanego powiadomienia co 30 minut (łącznie 5 razy) do grupy „MySQL Administrators". Aby to skonfigurować:

- Na karcie *Operations* ustaw *Default operation step duration* na „30m" (30 minut).
- Ustaw *Steps* eskalacji od „1" do „5".
- Wybierz grupę „MySQL Administrators" jako odbiorców wiadomości.



Powiadomienia będą wysyłane po 0:00, 0:30, 1:00, 1:30 i 2:00 godziny od wystąpienia problemu (chyba że problem zostanie oczywiście rozwiązany wcześniej).

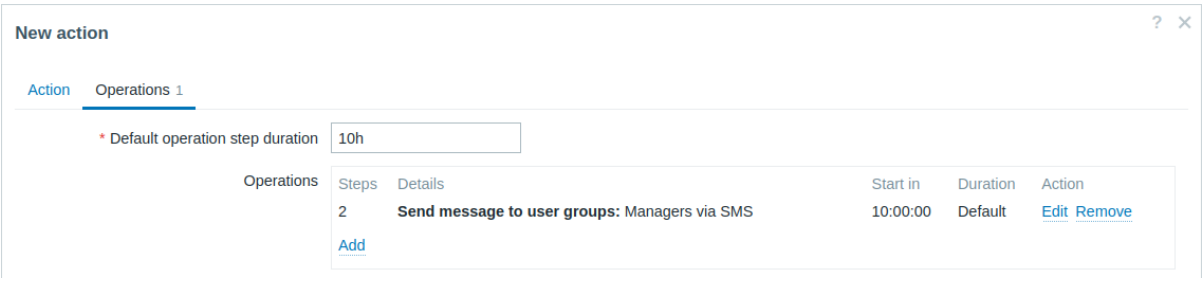
Jeśli problem zostanie rozwiązany i skonfigurowano wiadomość o odzyskaniu, zostanie ona wysłana do tych, którzy otrzymali co najmniej jedną wiadomość o problemie w ramach tego scenariusza eskalacji.

Note:
Jeśli wyzwalacz, który wygenerował aktywną eskalację, zostanie wyłączony, Zabbix wyśle wszystkim osobom, które otrzymały już powiadomienia, wiadomość informacyjną na ten temat.

Przykład 2

Wysyłanie opóźnionego powiadomienia o długotrwałym problemie. Aby to skonfigurować:

- Na karcie *Operations* ustaw *Default operation step duration* na „10h" (10 godzin).
- Ustaw *Steps* eskalacji od „2" do „2".



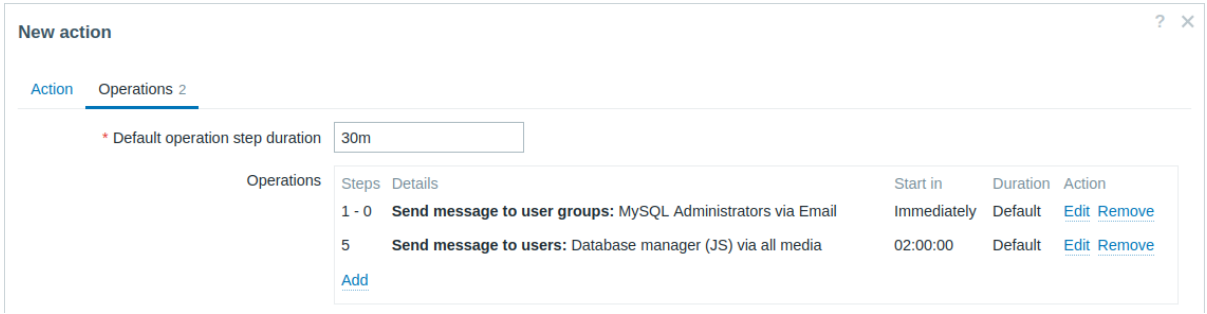
Powiadomienie zostanie wysłane dopiero w kroku 2 scenariusza eskalacji, czyli 10 godzin po wystąpieniu problemu.

Możesz dostosować treść wiadomości do czegoś w rodzaju „Problem trwa dłużej niż 10 godzin".

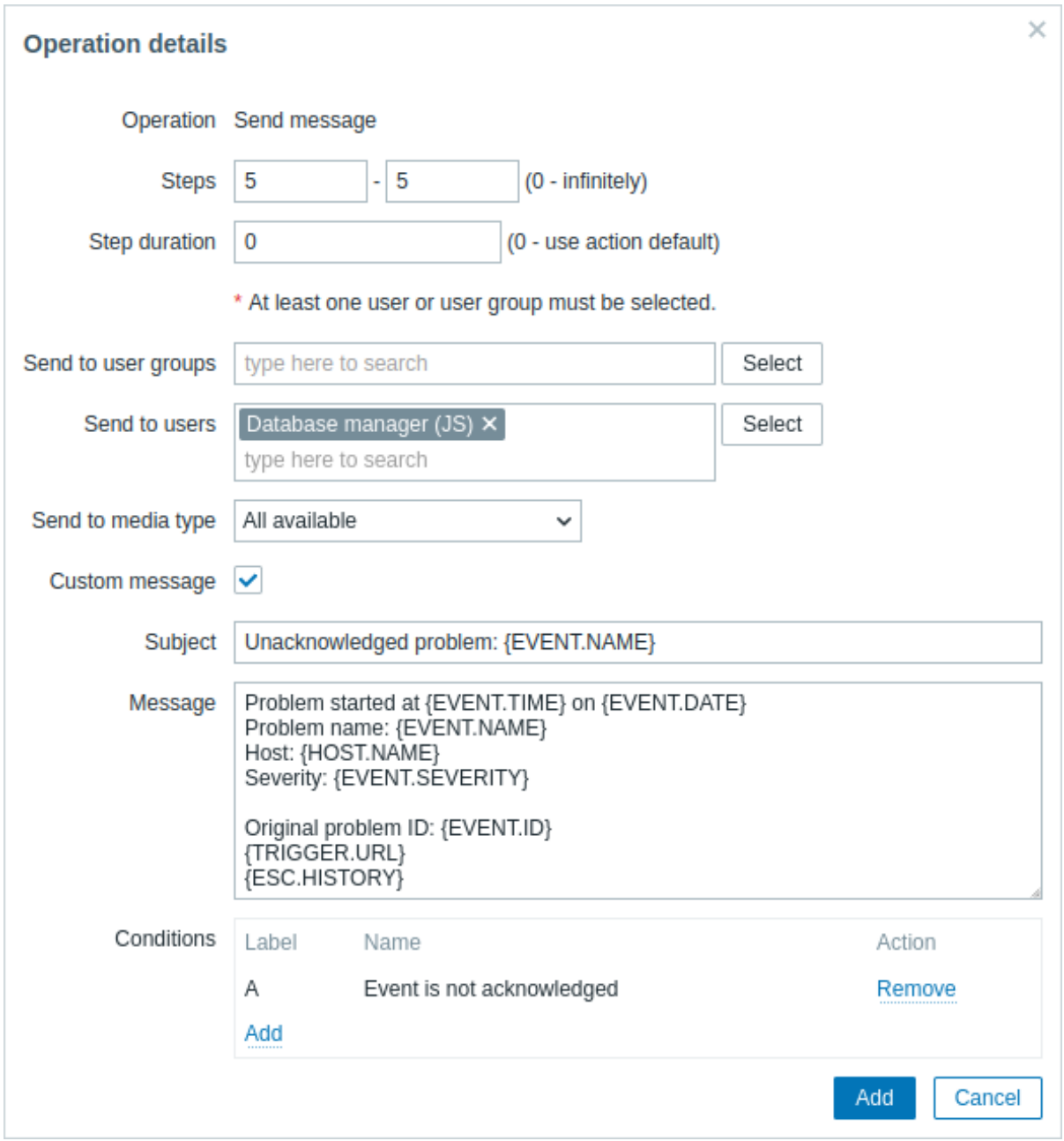
Przykład 3

Eskalacja problemu do Szefa.

W pierwszym przykładzie powyżej skonfigurowaliśmy okresowe wysyłanie wiadomości do administratorów MySQL. W tym przypadku administratorzy otrzymają cztery wiadomości, zanim problem zostanie eskalowany do menedżera bazy danych. Należy pamiętać, że menedżer otrzyma wiadomość tylko wtedy, gdy problem nie został jeszcze potwierdzony, co oznacza, że prawdopodobnie nikt jeszcze nad nim nie pracuje.



Szczegóły operacji 2:



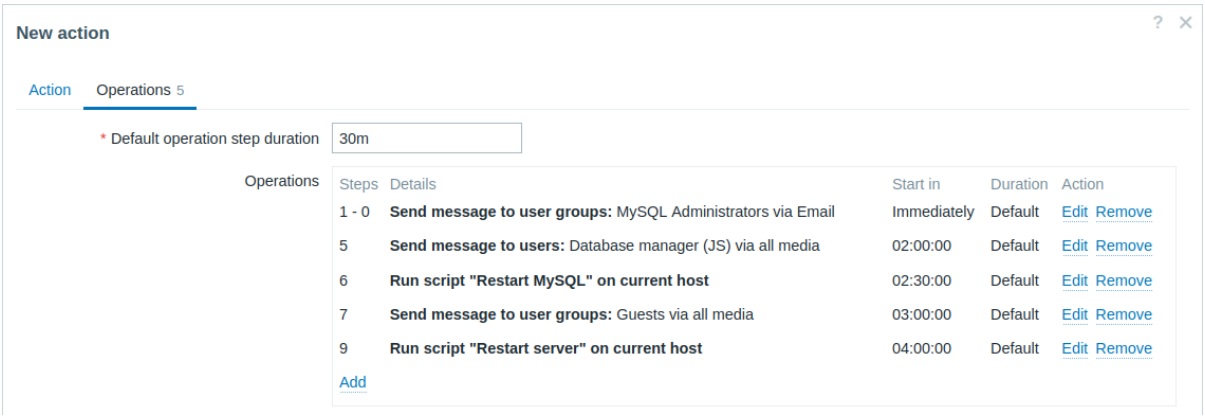
Zwróć uwagę na użycie makra {ESC.HISTORY} w dostosowanej wiadomości. Makro to będzie zawierać informacje o wszystkich wcześniej wykonanych krokach tej eskalacji, takich jak wysłane powiadomienia i wykonane polecenia.

Przykład 4

Bardziej złożony scenariusz. Po wysłaniu wielu wiadomości do administratorów MySQL i eskalacji do menedżera, Zabbix spróbuje ponownie uruchomić bazę danych MySQL. Stanie się to, jeśli problem będzie występował przez 2:30 godziny i nie zostanie potwierdzony.

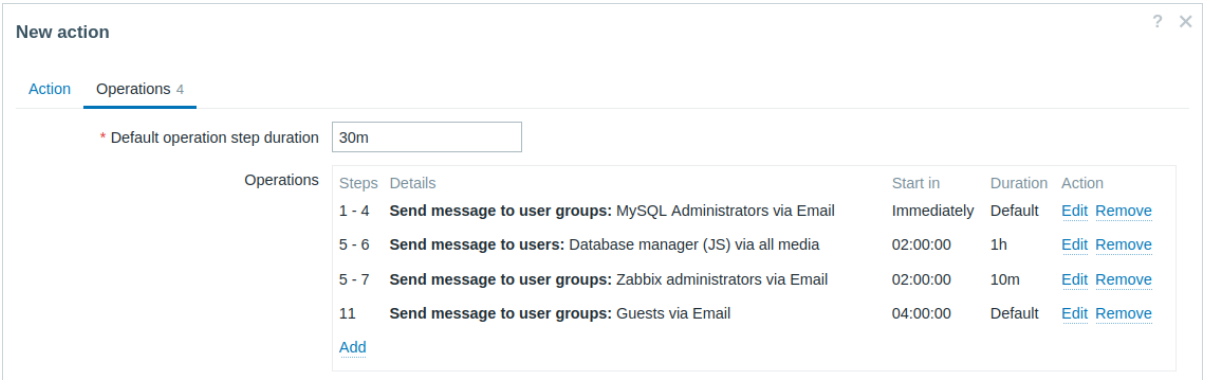
Jeśli problem nadal będzie występował, po kolejnych 30 minutach Zabbix wyśle wiadomość do wszystkich użytkowników gościnnych.

Jeśli to nie pomoże, po kolejnej godzinie Zabbix zrestartuje serwer z bazą danych MySQL (drugie zdalne polecenie) przy użyciu poleceń IPMI.



Przykład 5

Eskalacja z kilkoma operacjami, które mają nakładające się zakresy kroków i niestandardowe interwały. Domyślny czas trwania kroku operacji wynosi 30 minut.



Powiadomienia będą wysyłane w następujący sposób:

- Do administratorów MySQL o 0:00, 0:30, 1:00 i 1:30 od momentu wystąpienia problemu.
- Do menedżera bazy danych o 2:00 i 2:10 (krótszy niestandardowy czas trwania kroku wynoszący 10 minut, zdefiniowany w kolejnej operacji, zastępuje dłuższy czas trwania kroku wynoszący 1 godzinę skonfigurowany tutaj, zgodnie z opisem w Szczegóły operacji dla *Czas trwania kroku*, gdy kroki nakładają się).
- Do administratorów Zabbix o 2:00, 2:10 i 2:20 od momentu wystąpienia problemu (zastosowany jest niestandardowy czas trwania kroku wynoszący 10 minut).
- Do użytkowników guest o 4:00 od momentu wystąpienia problemu (domyślny czas trwania kroku wynoszący 30 minut obowiązuje między krokami 8 i 11).

Otrzymywanie powiadomień o nieobsługiwanych pozycjach

Przegląd

W Zabbix można otrzymywać powiadomienia o nieobsługiwanych pozycjach.

Jest to część koncepcji zdarzeń wewnętrznych w Zabbix, która umożliwia użytkownikom otrzymywanie powiadomień w takich sytuacjach. Zdarzenia wewnętrzne odzwierciedlają zmianę stanu:

- gdy pozycje przechodzą ze stanu „normalny” do „nieobsługiwany” (i z powrotem);
- gdy wyzwalacze przechodzą ze stanu „normalny” do „nieznany” (i z powrotem);
- gdy reguły niskopoziomowego wykrywania przechodzą ze stanu „normalny” do „nieobsługiwany” (i z powrotem).

W tej sekcji przedstawiono instrukcję dotyczącą **otrzymywania powiadomień**, gdy pozycja staje się nieobsługiwana.

Konfiguracja

Ogólnie proces konfigurowania powiadomienia powinien być znajomy dla osób, które wcześniej konfigurowały alerty w Zabbix.

Krok 1

Skonfiguruj wybrane medium, takie jak e-mail, SMS lub skrypt, które będzie używane do powiadomień. Aby wykonać to zadanie, zapoznaj się z odpowiednimi sekcjami instrukcji.

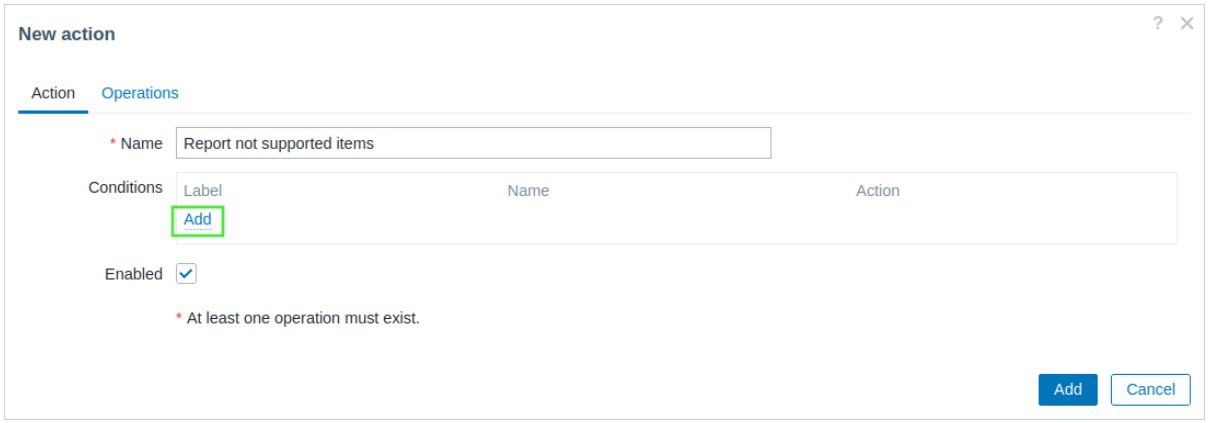
Attention:
Do powiadamiania o zdarzeniach wewnętrznych używana jest domyślna ważność („Niesklasyfikowane”), więc pozostaw ją zaznaczoną podczas konfigurowania mediów użytkownika, jeśli chcesz otrzymywać powiadomienia o zdarzeniach wewnętrznych.

Krok 2

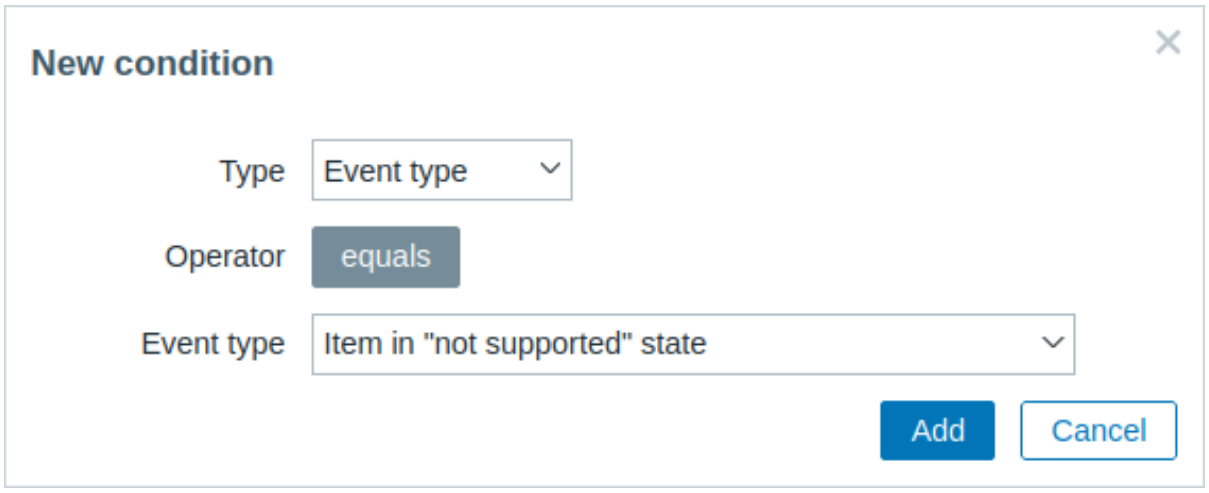
1. Przejdź do *Alerts* > *Actions* > *Internal actions*.
2. Kliknij *Create action* w prawym górnym rogu strony, aby otworzyć formularz konfiguracji akcji.

Krok 3

1. Na karcie *Action* wprowadź nazwę wyzwalacza.
2. Kliknij *Add* w bloku *Conditions*, aby dodać nowy warunek.



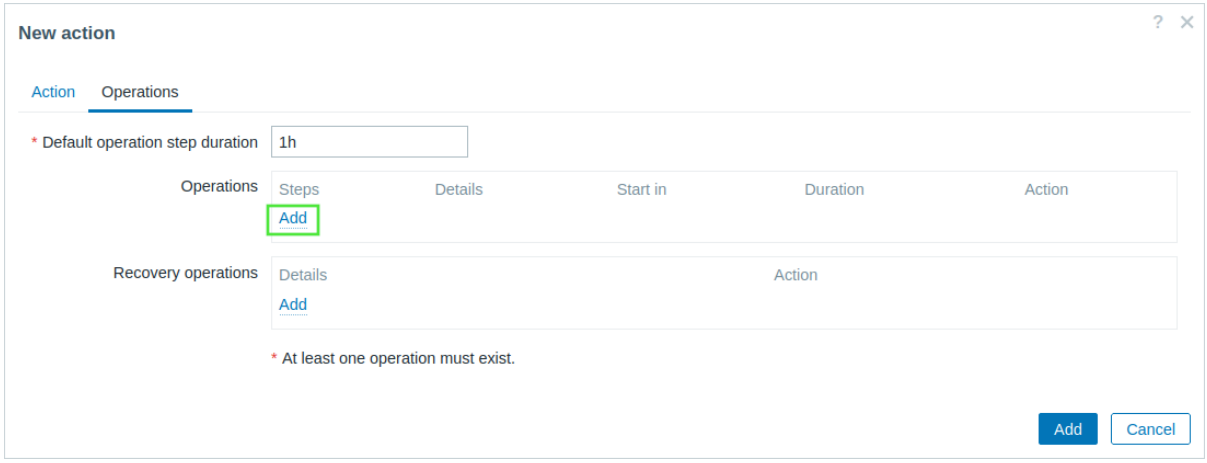
3. W wyskakującym oknie *New condition* wybierz „Event type” jako typ warunku, a następnie wybierz „Item in ’not supported’ state” jako typ zdarzenia.



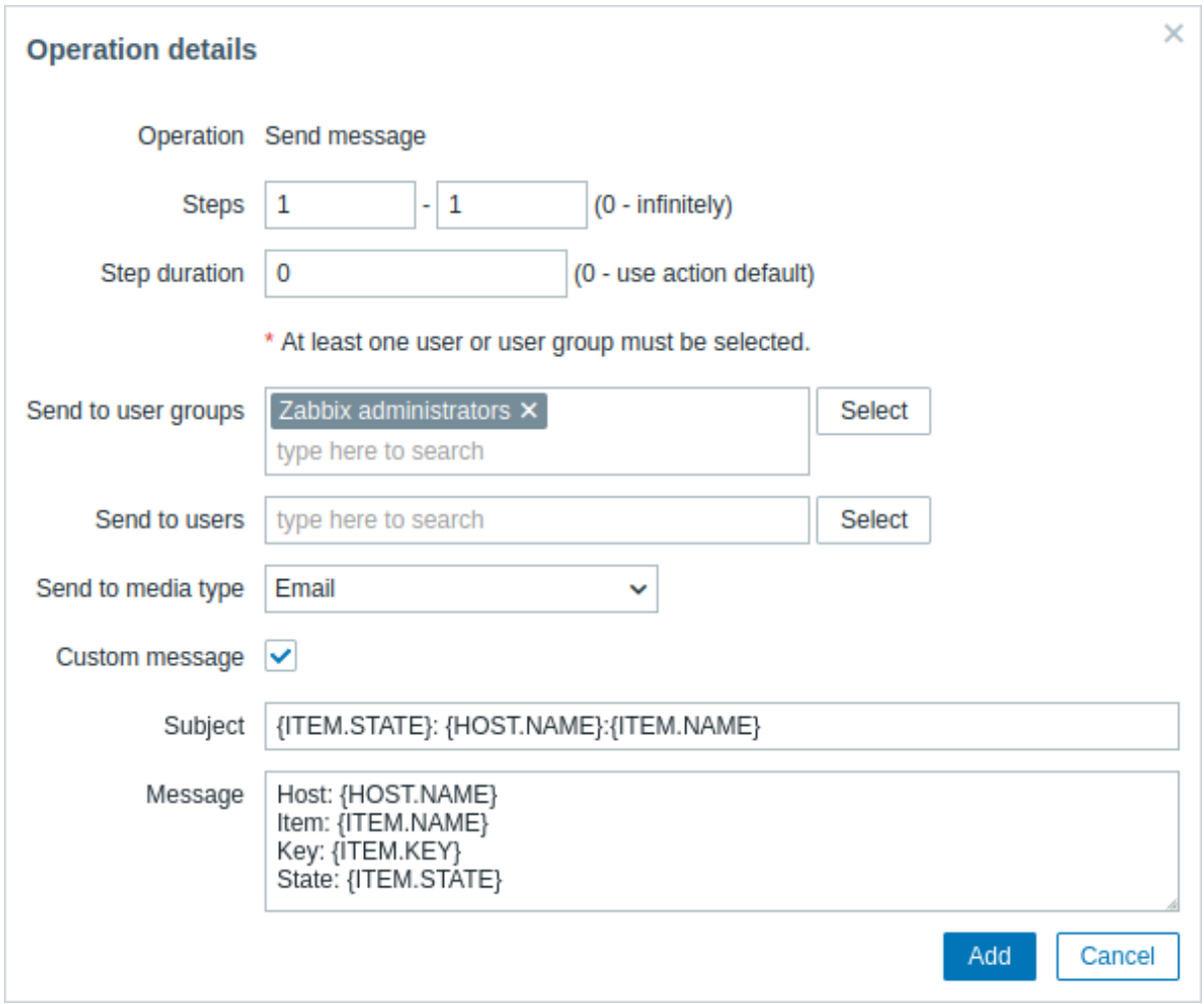
4. Kliknij *Add*, aby faktycznie wyświetlić warunek w bloku *Conditions*.

Krok 4

1. Na karcie *Operations* kliknij *Add* w bloku *Operations*, aby dodać nową operację.



2. Wybierz odbiorców wiadomości (grupy użytkowników/użytkowników) oraz typ nośnika (lub „All”), który ma być używany do dostarczenia wiadomości.
3. Zaznacz pole wyboru *Custom message*, jeśli chcesz wprowadzić niestandardowy temat/treść wiadomości o problemie.



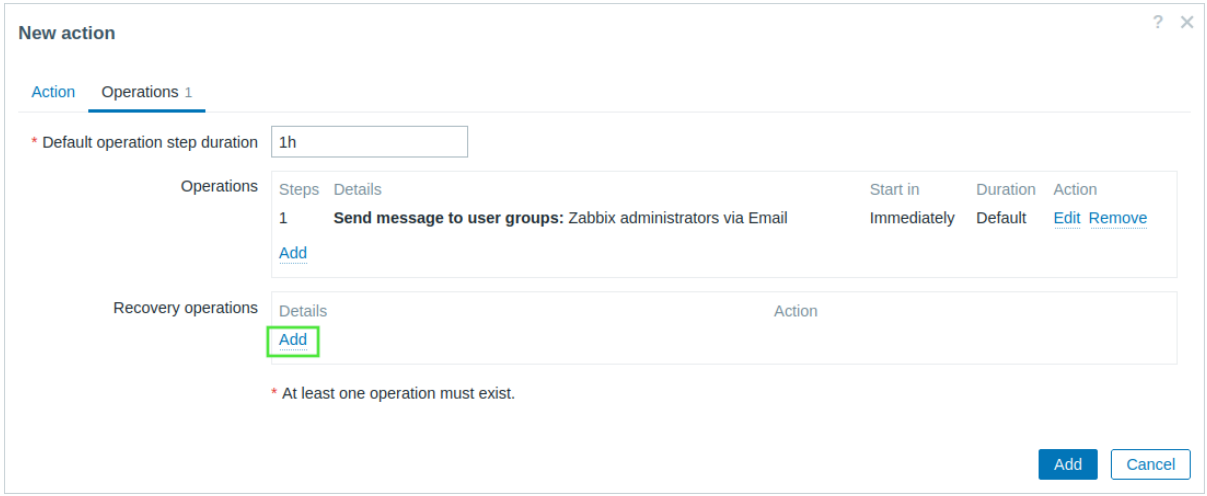
4. Kliknij *Add*, aby faktycznie dodać operację do bloku *Operations*.

Jeśli chcesz otrzymywać więcej niż jedno powiadomienie, ustaw czas trwania kroku operacji (odstęp między wysyłanymi wiadomościami) i dodaj kolejny krok.

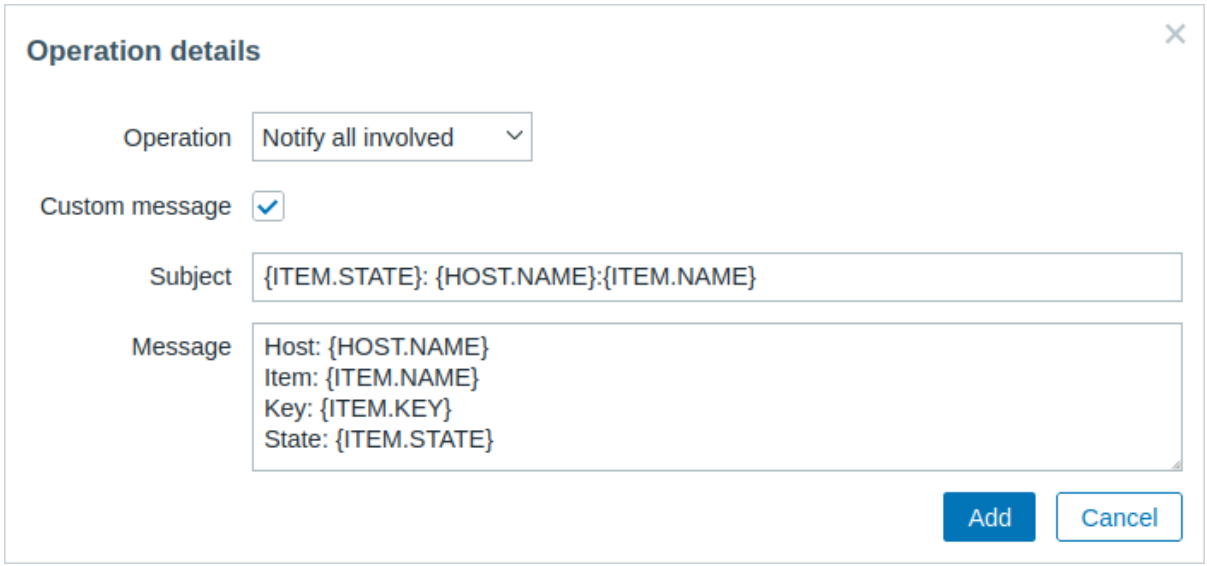
Krok 5

Blok *Operacje odzyskiwania* umożliwia skonfigurowanie powiadomienia o odzyskaniu, gdy pozycja powróci do normalnego stanu.

1. Kliknij *Dodaj* w bloku *Operacje odzyskiwania*, aby dodać nową operację odzyskiwania.



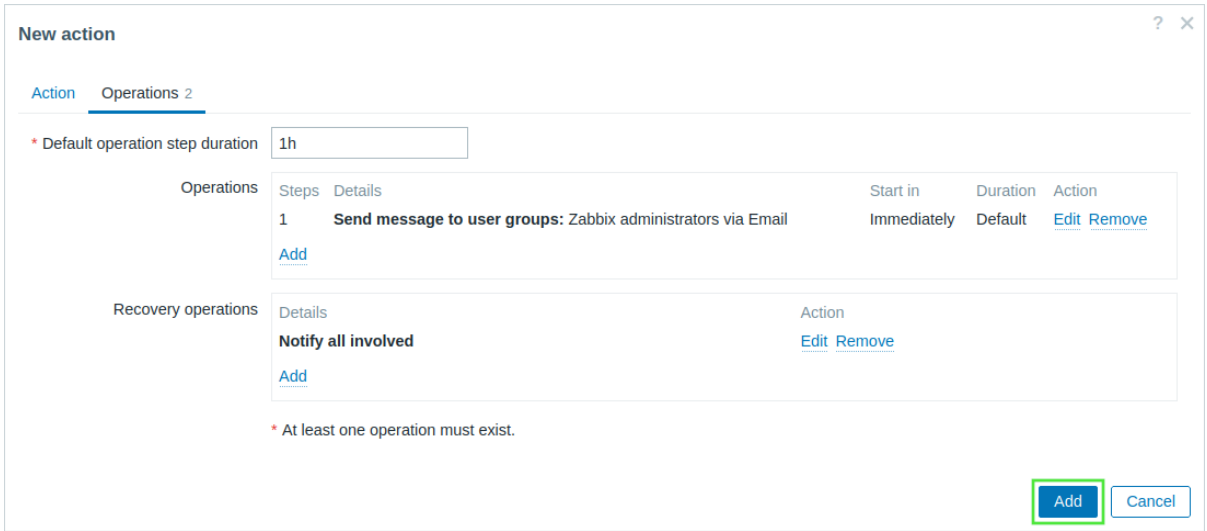
2. Wybierz typ operacji „Powiadom wszystkich zaangażowanych”.
3. Zaznacz pole wyboru *Niestandardowa wiadomość*, jeśli chcesz wprowadzić niestandardowy temat lub treść wiadomości o odzyskaniu.



4. Kliknij *Dodaj* w wyskakującym oknie *Szczegóły operacji*, aby faktycznie dodać operację do bloku *Operacje odzyskiwania*.

Krok 6

Po zakończeniu kliknij przycisk *Add* u dołu formularza.



I to wszystko, gotowe! Teraz możesz oczekiwać na otrzymanie pierwszego powiadomienia z Zabbix, jeśli jakaś pozycja stanie się nieobsługiwana.

Makra

Przegląd

Zabbix obsługuje szereg wbudowanych makr, których można używać w różnych sytuacjach. Te makra są zmiennymi identyfikowanymi za pomocą określonej składni:

`{MACRO}`

Makra są rozwijane do określonej wartości zależnie od kontekstu.

Efektywne korzystanie z makr pozwala oszczędzać czas i sprawia, że konfiguracja Zabbix jest bardziej przejrzysta.

W jednym z typowych zastosowań makro może być używane w szablonie. W związku z tym wyzwalacz w szablonie może mieć nazwę „Processor load is too high on {HOST.NAME}". Gdy szablon zostanie zastosowany do hosta, takiego jak serwer Zabbix, nazwa zostanie rozwinięta do „Processor load is too high on Zabbix server" podczas wyświetlania wyzwalacza w sekcji Monitoring.

Makra mogą być używane w parametrach klucza pozycji. Makro może być używane tylko w części parametru, na przykład `item.key[server_{HOST.HOST}_local]`. Ujęcie parametru w podwójne cudzysłowy nie jest konieczne, ponieważ Zabbix zajmie się wszelkimi niejednoznacznymi znakami specjalnymi, jeśli występują w rozwiniętym makrze.

W Zabbix istnieją także inne typy makr.

Zabbix obsługuje następujące makra:

- `{MACRO}` - wbudowane makro (zobacz pełną listę)
- `{<macro>.<func>(<params>)}` - funkcje makr
- `{$MACRO}` - makro zdefiniowane przez użytkownika, opcjonalnie z kontekstem
- `{#MACRO}` - makro dla wykrywania niskopoziomowego
- `{?EXPRESSION}` - makro wyrażenia

Funkcje makr

Omówienie

Funkcje makr umożliwiają dostosowywanie wartości makr (na przykład skracanie lub wyodrębnianie określonych podciągów), co ułatwia ich używanie.

Składnia funkcji makra jest następująca:

```
{macro.func(params)}
```

gdzie

- **macro** - makro do dostosowania;
- **func** - funkcja do zastosowania (zobacz obsługiwane funkcje);
- **params** - lista parametrów funkcji rozdzielona przecinkami, która musi być ujęta w **podwójne cudzysłowy**, jeśli:
  - zaczyna się od spacji lub podwójnego cudzysłowu;
  - zawiera zamykający nawias lub przecinek.

Na przykład:

```
{{TIME}.fmttime(format,time_shift)}
{{ITEM.VALUE}.regsub(pattern, output)}
{{$USERMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.regsub(pattern, output)}
```

Funkcje makr są obsługiwane dla

- makr wbudowanych
- makr użytkownika
- makr wykrywania niskiego poziomu
- makr wyrażeń

Funkcje makr mogą być używane we wszystkich lokalizacjach obsługujących wymienione makra. Dotyczy to sytuacji, chyba że wyraźnie zaznaczono, że oczekiwane jest wyłącznie makro (na przykład podczas konfigurowania makr hosta lub filtru reguły wykrywania niskiego poziomu).

Obsługiwana jest tylko jedna funkcja na makro; łączenie kilku funkcji makr w łańcuch nie jest obsługiwane.

Note:
Zobacz przykłady ucieczki znaków dla przypadków, gdy funkcje makr są używane wewnątrz innych kontekstów (funkcja, klucz pozycji, inne makro itp.).

Obsługiwane funkcje

Funkcje są wymienione bez dodatkowych informacji.
Kliknij funkcję, aby wyświetlić pełne informacje.

Funkcja	Opis
btoa	Kodowanie wartości makra w formacie Base64.
fmtnum	Formatowanie liczb w celu kontrolowania liczby cyfr wyświetlanych po przecinku dziesiętnym.
fmttime	Formatowanie czasu.
htmldecode	Dekodowanie wartości makra z kodowania HTML.
htmlencode	Kodowanie wartości makra w formacie HTML.
iregsub	Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (bez uwzględniania wielkości liter).
jsonpath	Zwracanie wyniku JSONPath.
lowercase	Przekształcanie znaków wartości makra na małe litery.
regrepl	Zastępowanie znaku/podciągu w wartości makra.
regsub	Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (z uwzględnieniem wielkości liter).
tr	Transliteracja znaków wartości makra.
uppercase	Przekształcanie znaków wartości makra na wielkie litery.
urldecode	Dekodowanie wartości makra z kodowania URL.
urlencode	Kodowanie wartości makra w formacie URL.
xmlxpath	Zwracanie wyniku XML XPath.

Szczegóły funkcji

Opcjonalne parametry funkcji są oznaczone przez < >.

btoa

Kodowanie wartości makra do kodowania Base64. Kodowanie Base64 to metoda przedstawiania danych binarnych jako tekstu, przydatna do przechowywania i bezpiecznego przesyłania zawartości binarnej za pośrednictwem protokołów tekstowych.

Przykład:

```
{{ITEM.VALUE}.btoa()} - zakoduje wartość taką jak "zabbix" do postaci Base64 "emFiYml4"
```

fmtnum(digits)

Formatowanie liczb umożliwiające kontrolę liczby cyfr wyświetlanych po przecinku.

Parametry:

- **digits** — liczba cyfr po przecinku. Prawidłowy zakres: 0-20. Końcowe zera nie będą dodawane.

Przykłady:

```
{{ITEM.VALUE}.fmtnum(2)} - zwróci "24.35" dla otrzymanej wartości "24.3483523"
{{ITEM.VALUE}.fmtnum(0)} - zwróci "24" dla otrzymanej wartości "24.3483523"
```

fmttime(format,<time_shift>)

Formatowanie czasu.
 Należy pamiętać, że tej funkcji można używać z makrami, które rozwiązują się do wartości w jednym z następujących formatów czasu:

- `hh:mm:ss`
- `yyyy-mm-ddThh:mm:ss[tz]` (standard ISO8601)
- znacznik czasu UNIX

Parametry:

- **format** - obowiązkowy ciąg formatu, zgodny z formatowaniem funkcji `strftime`;
- **time_shift** (opcjonalny) - przesunięcie czasu zastosowane do czasu przed formatowaniem; powinno zaczynać się od `-<N><time_unit>` lub `+<N><time_unit>`, gdzie:
  - `N` - liczba jednostek czasu do dodania lub odjęcia;

- `time_unit` - h (godzina), d (dzień), w (tydzień), M (miesiąc) lub y (rok).

Uwagi:

- Parametr `time_shift` obsługuje wieloetapowe operacje na czasie i może zawierać `/<time_unit>` do przesunięcia na początek jednostki czasu (`/d` - północ, `/w` - 1. dzień tygodnia (poniedziałek), `/M` - 1. dzień miesiąca itd.). Przykłady: `-1w` - dokładnie 7 dni wstecz; `-1w/w` - poniedziałek poprzedniego tygodnia; `-1w/w+1d` - wtorek poprzedniego tygodnia.
- Operacje na czasie są obliczane od lewej do prawej bez priorytetów. Na przykład `-1M/d+1h/w` zostanie zinterpretowane jako `((-1M/d)+1h)/w`.

Przykłady:

```
{{TIMESTAMP}.fmttime(%B)} - zwróci "October" z otrzymanej wartości "1633098961"
{{TIMESTAMP}.fmttime(%d %B,-1M/M)} - zwróci "1 September" z otrzymanej wartości "1633098961"
{{EVENT.TIMESTAMP}.fmttime("%x %X")} - zwróci reprezentację daty i czasu zależną od ustawień regionalnych;
{{EVENT.TIME}.fmttime("%x %X")} - zwróci reprezentację daty i czasu zależną od ustawień regionalnych; na p
```

htmldecode

Dekodowanie wartości makra z kodowania HTML.

Obsługiwane są następujące znaki:

Wartość	Zdekodowana wartość
&	&
<	<
>	>
"	"
'	'
'	'

Przykład:

```
{{ITEM.VALUE}.htmldecode()} - zdekoduje HTML wartość taką jak "&lt;" do "<"
```

htmlencode

Kodowanie wartości makra do kodowania HTML.

Obsługiwane są następujące znaki:

Wartość	Zakodowana wartość
&	&
<	<
>	>
"	"
'	'

Przykład:

```
{{ITEM.VALUE}.htmlencode()} - zakoduje znak HTML, taki jak "<", do postaci "&lt;"
```

iregsub(pattern,output)

Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (bez rozróżniania wielkości liter).

Parametry:

- **pattern** - wyrażenie regularne do dopasowania;
- **output** - opcje wyjściowe. Symbole zastępcze **\1 - \9** są obsługiwane dla grup przechwytywania. **\0** zwraca dopasowany tekst.

Uwagi:

- Jeśli nie ma dopasowania dla wyrażenia regularnego, funkcja zwraca pusty ciąg.
- Jeśli wzorzec funkcji jest niepoprawnym wyrażeniem regularnym, makro jest obliczane jako 'UNKNOWN' (z wyjątkiem makr wykrywania niskiego poziomu, w których przypadku funkcja zostanie zignorowana, a makro pozostanie nierozwiązane).
- Odwołania do nieistniejących grup przechwytywania w ciągu zastępczym są zastępowane pustym ciągiem.

Przykład:

```
{{ITEM.VALUE}.iregsub("fail|error|fault|problem","ERROR")} - zostanie rozwinięte do "ERROR", jeśli zostaną
```

jsonpath(pattern,<default>)

Zwraca wynik JSONPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **pattern** - wymagany wzorzec;
- **default** - opcjonalna wartość zastępcza, jeśli zapytanie JSONPath nie zwróci danych. Należy pamiętać, że w przypadku innych błędów zwracany jest błąd.

Uwagi:

- Funkcja kończy działanie błędem, jeśli:
  - kod JSON jest nieprawidłowy;
  - zapytanie JSONPath zakończy się niepowodzeniem;
  - wynik JSONPath nie zawiera danych, a parametr `default` nie został podany.

Przykład:

```
{{ITEM.VALUE}.jsonpath("$.columns.servicename", "NA")}
```

małe litery

Przekształcenie wszystkich znaków wartości makra na małe litery.
Działa z jednobajtowymi zestawami znaków (takimi jak ASCII) i nie obsługuje UTF-8.

Przykład:

```
{{ITEM.VALUE}.lowercase()} - przekształci wartość taką jak "Zabbix SERVER" na "zabbix server" (małe litery
```

regrepl(pattern,replacement,<pattern2>,<replacement2>,...)

Zastępuje znak/podciąg w wartości makra.

Parametry:

- **pattern** - wyrażenie regularne do dopasowania;
- **replacement** - ciąg zastępujący. **\1 - \9** są obsługiwane jako symbole zastępcze w ciągach zastępujących, aby przechwytywać grupy.

Komentarze:

- Wszystkie dopasowania zgodne ze wzorcem są zastępowane.
- Wzorce i zamienniki są przetwarzane sekwencyjnie, przy czym każda kolejna para jest stosowana zgodnie z wynikiem poprzedniego zastąpienia.
- Odwołania do nieistniejących grup przechwytywania w ciągu zastępującym są zastępowane pustym ciągiem.

Przykłady:

```
{{ITEM.VALUE}.regrepl("oldParam", "newParam")} - zastąpi "oldParam" ciągiem "newParam"
{{ITEM.VALUE}.regrepl("([^a-z])","\\\1")} - wszystkie znaki inne niż litery zostaną poprzedzone ukośnikiem
{$THRESHOLD:"{{#FSNAME}.regrepl(\"\\$\",\"\")}"} - usunie końcowy ukośnik odwrotny (na przykład, aby zastą
{{ITEM.VALUE}.regrepl("_v1\.0", "_v2.0", "\(final\)", "")} - zastąpi wiele fragmentów w wartości pozycji
```

regsub(pattern,output)

Wyodrębnianie podciągu na podstawie dopasowania wyrażenia regularnego (z uwzględnieniem wielkości liter).

Parametry:

- **pattern** - wyrażenie regularne do dopasowania;
- **output** - opcje danych wyjściowych. Symbole zastępcze **\1 - \9** są obsługiwane dla grup przechwytywania. **\0** zwraca dopasowany tekst.

Uwagi:

- Jeśli nie ma dopasowania do wyrażenia regularnego, funkcja zwraca pusty ciąg.
- Jeśli wzorzec funkcji jest niepoprawnym wyrażeniem regularnym, makro przyjmuje wartość 'UNKNOWN' (z wyjątkiem makr wykrywania niskiego poziomu, w których przypadku funkcja zostanie zignorowana, a makro pozostanie nierozwiązane).
- Odwołania do nieistniejących grup przechwytywania w ciągu zastępczym są zastępowane pustym ciągiem.

Przykłady:

```
{{ITEM.VALUE}.regsub("^([0-9]+)", Problem ID: \1)} - zostanie rozwinięte do "Problem ID: 123", jeśli zosta
{{ITEM.VALUE}.regsub("fail|error|fault|problem","ERROR")} - zostanie rozwinięte do "ERROR", jeśli zostaną
```

Zobacz więcej przykładów.

tr(characters,replacement)

Transliteracja znaków wartości makra.

- **characters** - zestaw znaków do zastąpienia;
- **replacement** - zestaw odpowiadających im pozycyjnie znaków zastępczych.

Przykłady:

```
{{ITEM.VALUE}.tr(abc, xyz)} - zastąpi wszystkie wystąpienia "a" przez "x", "b" przez "y", "c" przez "z"
{{ITEM.VALUE}.tr(abc, xyzq)} - zastąpi wszystkie wystąpienia "a" przez "x", "b" przez "y", "c" przez "z" (
{{ITEM.VALUE}.tr(abcde, xyz)} - zastąpi wszystkie wystąpienia "a" przez "x", "b" przez "y", "c" przez "z",
{{ITEM.VALUE}.tr("\\\'", "\/\"")} - zastąpi wszystkie wystąpienia ukośnika odwrotnego ukośnikiem prostym,
{{ITEM.VALUE}.tr(A-Z,a-z)} - przekonwertuje wszystkie litery na małe
{{ITEM.VALUE}.tr(0-9a-z,*)} - zastąpi wszystkie cyfry i małe litery znakiem "*"
{{ITEM.VALUE}.tr(0-9,ab)} - zastąpi wszystkie wystąpienia 0 przez "a", a wszystkie wystąpienia 1, 2, 3, 4,
{{ITEM.VALUE}.tr(0-9abcA-L,*)} - zastąpi wszystkie cyfry, znaki "abc" oraz zakres A-L znakiem "*"
{{ITEM.VALUE}.tr("\n","*")} - zastąpi wystąpienia końca linii znakiem *
{{ITEM.VALUE}.tr("e", "\n")} - zastąpi wszystkie "e" znakiem końca linii
```

Aby uwzględnić znaki literałowe:

```
ukośnik odwrotny - musi być zapisany jako \\
pojedynczy cudzysłów - musi być zapisany jako \'
podwójny cudzysłów - musi być zapisany jako \"
```

Obsługiwane sekwencje ucieczki z ukośnikiem odwrotnym:

```
\\\\ => \\ - podwójny ukośnik odwrotny na pojedynczy ukośnik odwrotny
\\a  => \a - alert
\\b  => \b - backspace
\\f  => \f - wysunięcie strony
\\n  => \n - nowa linia
\\r  => \r - powrót karetki
\\t  => \t - tabulator poziomy
\\v  => \v - tabulator pionowy
```

wielkie litery

Przekształcenie wszystkich znaków wartości makra na wielkie litery. Działa z jednobajtowymi zestawami znaków (takimi jak ASCII) i nie obsługuje UTF-8.

Przykład:

```
{{ITEM.VALUE}.uppercase()} - przekształci wartość taką jak "Zabbix Server" na "ZABBIX SERVER" (wielkie lit
```

urldecode

Dekodowanie wartości makra z kodowania URL.

Przykład:

```
{{ITEM.VALUE}.urldecode()} - zdekoduje URL wartość taką jak "%2F" do "/"
```

urlencode

Kodowanie wartości makra do formatu URL encoding.

Przykład:

```
{{ITEM.VALUE}.urlencode()} - zakoduje znak taki jak "/" do postaci "%2F"
```

xmlxpath(pattern,<default>)

Zwraca wynik XML XPath.
 Obsługiwane typy wartości: *String*, *Text*, *Log*.

Parametry:

- **pattern** - wymagany wzorzec;
- **default** - opcjonalna wartość zastępcza, jeśli zapytanie XPath nie zwróci danych. Należy pamiętać, że w przypadku innych błędów zwracany jest stan niepowodzenia.

Uwagi:

- Funkcja kończy działanie błędem, jeśli:
  - XML jest nieprawidłowy.
  - Zapytanie XPath zakończy się niepowodzeniem.
  - Wynik XPath nie zawiera danych, a parametr `default` nie został podany.
  - Zabbix został skompilowany bez obsługi XML.

Przykład:

```
{{ITEM.VALUE}.xmlxpath("/response/error/status","unknown")}
```

Dodatkowe przykłady

Poniższa tabela pokazuje więcej przykładów użycia funkcji makr.

`{#IFALIAS}` jest makrem LLD i jest zdefiniowane tylko w kontekstach low-level discovery (reguły wykrywania, prototypy oraz pozycje/wyzwalacze utworzone na ich podstawie). Użycie go poza LLD spowoduje pozostawienie tokenu bez rozwinięcia.

Funkcja makra	Otrzymana wartość	Wynik
`{{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*` (nieprawidłowe wyrażenie regularne)
`{{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)}`	`customername_1`	`customername`
`{{#IFALIAS}.regsub("(.*)_([0-9]+)", \2)}`	`customername_11`	
`{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)}`	`customername_1`	`{{#IFALIAS}.regsub("(.*)_([0-9]+", \1)}` (nieprawidłowe wyrażenie regularne)
`{$MACRO:"{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}"}`	`customername_1`	`{$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)}"}` (nieprawidłowe wyrażenie regularne)
`"{$MACRO:"\{{#IFALIAS}.regsub("(.*)_([0-9]+)", \1)}\"}"`	`customername_1`	`"{$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)}\"}"` (nieprawidłowe wyrażenie regularne)

Wyświetlanie pełnych wartości pozycji

Długie wartości rozwiniętych makr {ITEM.VALUE} i {ITEM.LASTVALUE} dla pozycji tekstowych/logów są w niektórych miejscach frontend skracane do 20 znaków. Aby zobaczyć pełne wartości tych makr, można użyć funkcji makr, np.:

```
{{ITEM.VALUE}.regsub("(.*)", \1)}
{{ITEM.LASTVALUE}.regsub("(.*)", \1)}
```

Zobacz także: szczegóły makr {ITEM.VALUE} i {ITEM.LASTVALUE}.

Makra użytkownika

Omówienie

Makra użytkownika są obsługiwane w Zabbix w celu zapewnienia większej elastyczności, oprócz makr obsługiwanych standardowo.

Makra użytkownika można definiować na poziomie globalnym, szablonu i hosta. Makra te mają specjalną składnię:

```
{$MACRO}
```

Zabbix rozwiązuje makra zgodnie z następującym priorytetem:

1. Makra na poziomie hosta (sprawdzane jako pierwsze).
2. Makra zdefiniowane dla szablonów pierwszego poziomu hosta (tj. szablonów bezpośrednio powiązanych z hostem), posortowane według identyfikatora szablonu.
3. Makra zdefiniowane dla szablonów drugiego poziomu hosta, posortowane według identyfikatora szablonu.
4. Makra zdefiniowane dla szablonów trzeciego poziomu hosta, posortowane według identyfikatora szablonu itd.
5. Makra globalne (sprawdzane jako ostatnie).

Innymi słowy, jeśli makro nie istnieje dla hosta, Zabbix spróbuje znaleźć je w szablonach hosta, zwiększając poziom zagnieżdżenia. Jeśli nadal nie zostanie znalezione, zostanie użyte makro globalne, jeśli istnieje.

Warning:
Jeśli makro o **tej samej nazwie** istnieje w wielu powiązanych szablonach tego samego poziomu, zostanie użyte makro z szablonu o najniższym identyfikatorze. Dlatego używanie makr o tej samej nazwie w wielu szablonach stanowi ryzyko związane z konfiguracją.

Jeśli Zabbix nie może znaleźć makra, makro nie zostanie rozwiązane.

Attention:
Makra (w tym makra użytkownika) pozostają nierozwiązane w sekcji Konfiguracja (na przykład na liście wyzwalaczy), aby zgodnie z założeniami ułatwić przejrzyste przedstawianie złożonej konfiguracji.

Makra użytkownika mogą być używane w:

- nazwie pozycji
- parametrze klucza pozycji
- interwałach aktualizacji pozycji i interwałach elastycznych
- nazwie i opisie wyzwalacza
- parametrach i stałych wyrażenia wyzwalacza (zobacz przykłady)
- wielu innych miejscach — zobacz pełną listę

Typowe przypadki użycia makr globalnych i makr hosta

- użycie makra globalnego w kilku miejscach; następnie zmiana wartości makra i zastosowanie zmian konfiguracji we wszystkich miejscach jednym kliknięciem
- wykorzystanie szablonów z atrybutami specyficznymi dla hosta: hasła, numery portów, nazwy plików, wyrażenia regularne itp.

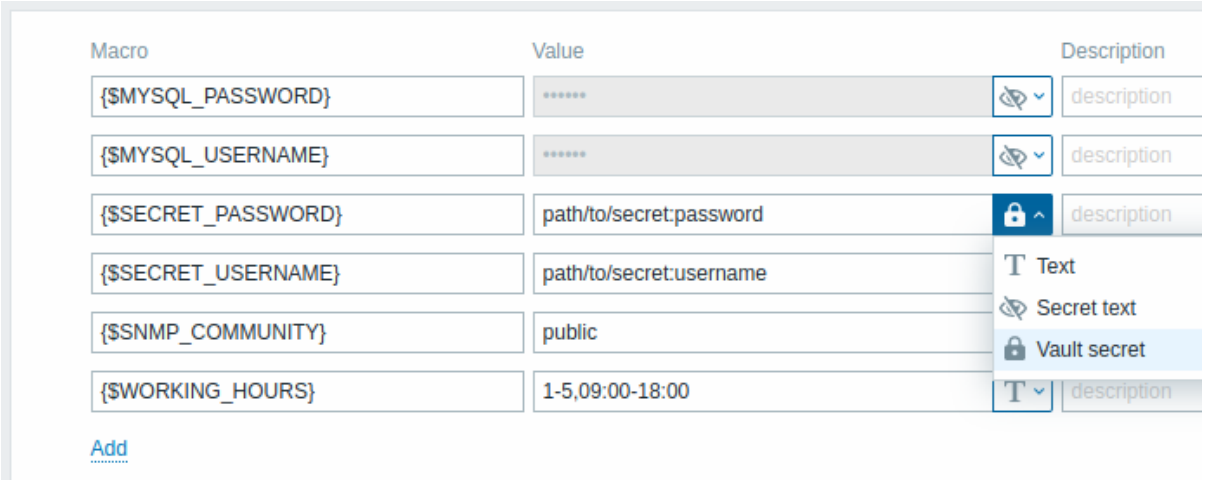
Note:
Zaleca się używanie makr hosta zamiast makr globalnych, ponieważ dodawanie, aktualizowanie lub usuwanie makr globalnych wymusza przyrostową aktualizację konfiguracji dla wszystkich hostów. Więcej informacji można znaleźć w sekcji Passive and active agent checks.

Konfiguracja

Aby zdefiniować makra użytkownika, przejdź do odpowiedniej lokalizacji w frontend:

- Dla makr globalnych wybierz *Administracja* > *Makra*.
- Dla makr na poziomie hosta i szablonu otwórz właściwości hosta lub szablonu i znajdź kartę *Makra*.

Makro użytkownika ma następujące atrybuty:



Parameter	Description
Macro	Nazwa makra. Nazwa musi być ujęta w nawiasy klamrowe i zaczynać się od znaku dolara. Przykład: `{$FRONTEND_URL}`. W nazwach makr dozwolone są następujące znaki: **A-Z** (tylko wielkie litery), **0-9**, _, **.**
Value	Wartość makra. Obsługiwane są trzy typy wartości: **Tekst** (domyślnie) - wartość w postaci zwykłego tekstu **Tajny tekst** - wartość jest maskowana gwiazdkami **Sekret z vault** - wartość zawiera ścieżkę/zapytanie do sekretu z vault. Aby zmienić typ wartości, kliknij przycisk na końcu pola wprowadzania wartości. Maksymalna długość wartości makra użytkownika wynosi 2048 znaków.
Description	Pole tekstowe służące do podania dodatkowych informacji o tym makrze.

Podczas konfigurowania makr użytkownika **zwróć uwagę** na następujące zachowania zależne od kontekstu:

- jeśli makra użytkownika są używane w pozycjach szablonu lub wyzwalaczach, rozważ dodanie tych makr również do szablonu (nawet jeśli są zdefiniowane globalnie); w ten sposób makra typu *Tekst* będą działać zgodnie z oczekiwaniami po wyeksportowaniu szablonu do XML i zaimportowaniu go do innego systemu (wartości tajnych makr nie są eksportowane)
- jeśli makra użytkownika są używane w wyrażeniach wyzwalaczy, zostaną one rozwiązane tylko wtedy, gdy odwołują się do parametru lub stałej; NIE zostaną rozwiązane, jeśli odwołują się do hosta, klucza pozycji, funkcji, operatora lub innego wyrażenia wyzwalacza (tajnych makr nie można używać w wyrażeniach wyzwalaczy)
- jeśli makra użytkownika są używane na hoście, który ma regułę niskopoziomowego wykrywania z prototypami hostów, wykryte hosty odziedziczą wszystkie makra użytkownika zdefiniowane na tym hoście

Przykłady

Przykład 1

Użycie makra na poziomie hosta w kluczu pozycji „Status demona SSH”:

```
net.tcp.service[ssh,,{$SSH_PORT}]
```

Tę pozycję można przypisać do wielu hostów, pod warunkiem że wartość `{$SSH_PORT}` jest zdefiniowana na tych hostach.

Przykład 2

Użycie makra na poziomie hosta w wyzwalaczu „Obciążenie procesora jest zbyt wysokie”:

```
last(/ca_001/system.cpu.load[,avg1])>{$MAX_CPULOAD}
```

Taki wyzwalacz zostałby utworzony w szablonie, a nie edytowany na poszczególnych hostach.

Note:
Jeśli chcesz użyć liczby wartości jako parametru funkcji (na przykład `max(/host/key,#3)`), uwzględnij znak hash w definicji makra, tak jak w tym przykładzie: `SOME_PERIOD => #3`.

Przykład 3

Użycie dwóch makr w wyzwalaczu „Obciążenie procesora jest zbyt wysokie”:

```
min(/ca_001/system.cpu.load[,avg1],{$CPULOAD_PERIOD})>{$MAX_CPULOAD}
```

Należy pamiętać, że makro może być używane jako parametr funkcji wyzwalacza, w tym przykładzie funkcji `min()`.

Przykład 4

Synchronizowanie warunku niedostępności agent z interwałem aktualizacji pozycji:

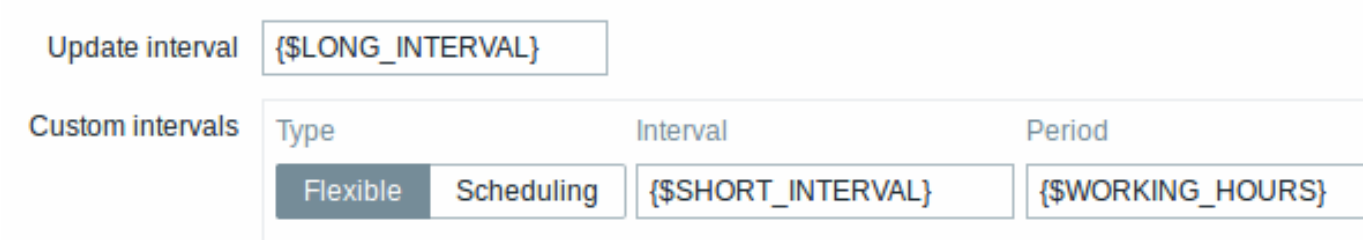
- zdefiniuj makro `{$INTERVAL}` i użyj go w interwale aktualizacji pozycji;
- użyj `{$INTERVAL}` jako parametru wyzwalacza niedostępności agent:

```
nodata(/ca_001/agent.ping,{$INTERVAL})=1
```

Przykład 5

Centralizacja konfiguracji godzin pracy:

- utwórz globalne makro `{$WORKING_HOURS}` o wartości `1-5,09:00-18:00`;
- użyj go w polu *Czas pracy* w sekcji *Administracja* > *Ogólne* > *GUI*;
- użyj go w polu *Kiedy aktywne* w sekcji *Użytkownicy* > *Użytkownicy*, na karcie *Media* użytkownika;
- użyj go do skonfigurowania częstszego odpytywania pozycji w godzinach pracy:

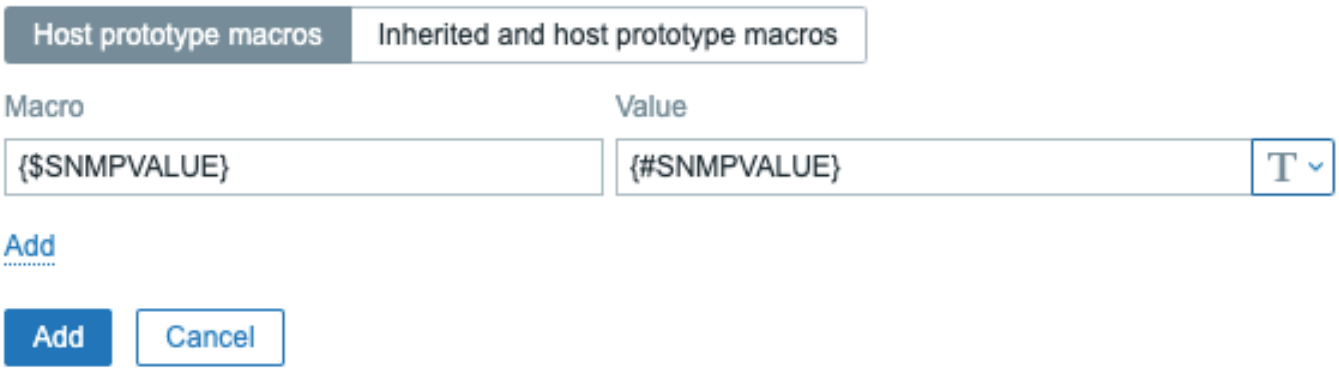


- użyj go w warunku działania *Okres*;
- w razie potrzeby dostosuj czas pracy w sekcji *Administracja* > *Makra*.

Przykład 6

Użyj makra prototypu hosta, aby skonfigurować pozycje dla wykrytych hostów:

1. W prototypie hosta zdefiniuj makro użytkownika `{$SNMPVALUE}`, używając makra wykrywania niskopoziomowego `{#SNMPVALUE}` jako wartości:



2. Przypisz szablon *Generic SNMPv2* do prototypu hosta.
3. Użyj `{$SNMPVALUE}` w polu *SNMP OID* pozycji szablonu *Generic SNMPv2*.

Kontekst makra użytkownika

Zobacz makra użytkownika z kontekstem.

Makra użytkownika z kontekstem

Omówienie

W makrach użytkownika można użyć opcjonalnego kontekstu, który umożliwia zastąpienie wartości domyślnej wartością właściwą dla danego kontekstu.

Kontekst jest dołączany do nazwy makra; składnia zależy od tego, czy kontekst jest statyczną wartością tekstową:

```
{$MACRO:"tekst statyczny"}
```

czy wyrażeniem regularnym:

`{$MACRO:regex:"wyrażenie regularne"}`

Należy pamiętać, że makro z kontekstem będącym wyrażeniem regularnym można definiować wyłącznie w konfiguracji makr użytkownika. Jeśli prefiks `regex:` zostanie użyty w innym miejscu jako kontekst makra użytkownika, na przykład w wyrażeniu wyzwalacza, zostanie potraktowany jako kontekst statyczny.

Cytowanie kontekstu jest opcjonalne (zobacz także ważne uwagi).

Przykłady kontekstu makra:

Przykład	Opis
`{$LOW_SPACE_LIMIT}`	Makro użytkownika bez kontekstu.
`{$LOW_SPACE_LIMIT:/tmp}`	Makro użytkownika z kontekstem (ciąg statyczny).
`{$LOW_SPACE_LIMIT:regex:"^/tmp$"}`	Makro użytkownika z kontekstem (wyrażenie regularne). Odpowiednik `{$LOW_SPACE_LIMIT:/tmp}`.
`{$LOW_SPACE_LIMIT:regex:"^/var/log/.*$"}`	Makro użytkownika z kontekstem (wyrażenie regularne). Dopasowuje wszystkie ciągi rozpoczynające się od /var/log/.

Przypadki użycia

Makra użytkownika z kontekstem mogą być definiowane w celu uzyskania bardziej elastycznych progów w wyrażeniach wyzwalaczy (na podstawie wartości pobieranych przez odkrywanie niskiego poziomu). Na przykład można zdefiniować następujące makra:

- {$LOW_SPACE_LIMIT} = 10
- {$LOW_SPACE_LIMIT:/home} = 20
- {$LOW_SPACE_LIMIT:regex:"^/[a-z]+$"} = 30

Następnie makro odkrywania niskiego poziomu może być użyte jako kontekst makra w prototypie wyzwalacza dla odkrywania zamontowanych systemów plików:

`last(/host/vfs.fs.size[{#FSNAME},pfree])<{$LOW_SPACE_LIMIT:"{#FSNAME}"}`

Po odkryciu będą stosowane różne progi niskiej ilości wolnego miejsca w wyzwalaczach, zależnie od wykrytych punktów montowania lub typów systemów plików. Zdarzenia problemu zostaną wygenerowane, jeśli:

- katalog /home ma mniej niż 20% wolnego miejsca na dysku
- katalogi pasujące do wzorca wyrażenia regularnego (takie jak /etc, /tmp lub /var) mają mniej niż 30% wolnego miejsca na dysku
- katalogi, które nie pasują do wzorca wyrażenia regularnego i nie są /home, mają mniej niż 10% wolnego miejsca na dysku

Ważne uwagi

- Jeśli istnieje więcej niż jedna makrodefinicja użytkownika z kontekstem, Zabbix najpierw spróbuje dopasować makra z prostym kontekstem, a następnie makra kontekstowe z wyrażeniami regularnymi w niezdefiniowanej kolejności.

Warning:
Nie twórz różnych makr kontekstowych pasujących do tego samego ciągu, aby uniknąć niezdefiniowanego zachowania.

- Jeśli makro wraz z jego kontekstem nie zostanie znalezione na hoście, w podłączonych szablonach ani globalnie, wyszukiwane jest makro bez kontekstu.
- W kontekście obsługiwane są tylko makra wykrywania niskiego poziomu. Wszelkie inne makra są ignorowane i traktowane jako zwykły tekst.

Technicznie rzecz biorąc, kontekst makra jest określany przy użyciu reguł podobnych do parametrów klucza pozycji, z wyjątkiem tego, że kontekst makra nie jest analizowany jako kilka parametrów, jeśli zawiera znak `,`:

- Kontekst makra musi być ujęty w " , jeśli kontekst zawiera znak } lub zaczyna się od znaku ". Cudzysłowy wewnątrz ujętego w cudzysłów kontekstu muszą być poprzedzone znakiem ucieczki \.
- Sam znak \ nie jest znakowany znakiem ucieczki, co oznacza, że niemożliwe jest użycie ujętego w cudzysłów kontekstu kończącego się znakiem \ — makro {$MACRO:"a:\b\c\"} jest nieprawidłowe.
- Początkowe spacje w kontekście są ignorowane, końcowe nie:
  - Na przykład {$MACRO:A} jest takie samo jak {$MACRO: A}, ale nie jak {$MACRO:A }.
- Wszystkie spacje przed początkowym cudzysłowem i po końcowym cudzysłowie są ignorowane, ale wszystkie spacje wewnątrz cudzysłowów nie:
  - Makra {$MACRO:"A"}, {$MACRO: "A"}, {$MACRO:"A" } oraz {$MACRO: "A" } są takie same, ale makra {$MACRO:"A"} i {$MACRO:" A "} nie są.

Wszystkie poniższe makra są równoważne, ponieważ mają ten sam kontekst: {$MACRO:A}, {$MACRO: A} oraz {$MACRO:"A"}. Stanowi to kontrast wobec kluczy pozycji, gdzie 'key[a]', 'key[a]' oraz 'key["a"]' są semantycznie takie same, ale różnią się pod względem unikalności.

Tajne makra użytkownika

Przegląd

Zabbix udostępnia dwie opcje ochrony poufnych informacji w wartościach makr użytkownika:

- Tajny tekst
- Sekret z vault

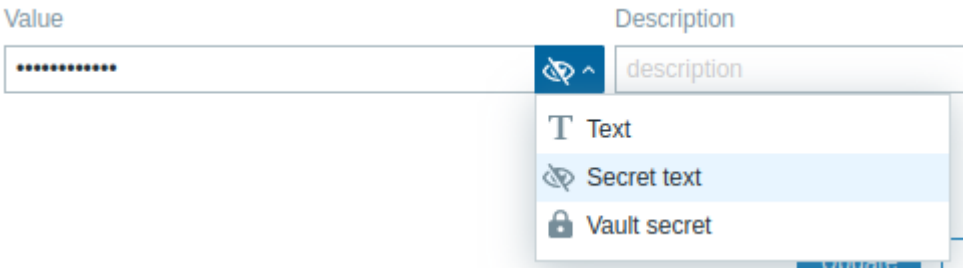
Note:
Chociaż wartość tajnego makra jest ukryta, może zostać ujawniona poprzez użycie w pozycjach. Na przykład w skrypcie zewnętrznym instrukcja echo odwołująca się do tajnego makra może zostać użyta do ujawnienia wartości makra we frontend, ponieważ serwer Zabbix ma dostęp do rzeczywistej wartości makra. Zobacz lokalizacje, w których wartości tajnych makr są odsłaniane.

Tajnych makr nie można używać w wyrażeniach wyzwalaczy.

Tajny tekst

W przypadku tajnych makr tekstowych wartość makra jest maskowana gwiazdkami.

Aby ustawić wartość makra jako tajną, kliknij przycisk na końcu pola *Value* i wybierz opcję *Secret text*:



Po zapisaniu konfiguracji nie będzie już możliwe wyświetlenie wartości.

Aby zmienić wartość makra, najedź kursorem na pole *Value* i kliknij przycisk *Set new value* (pojawia się po najechaniu):



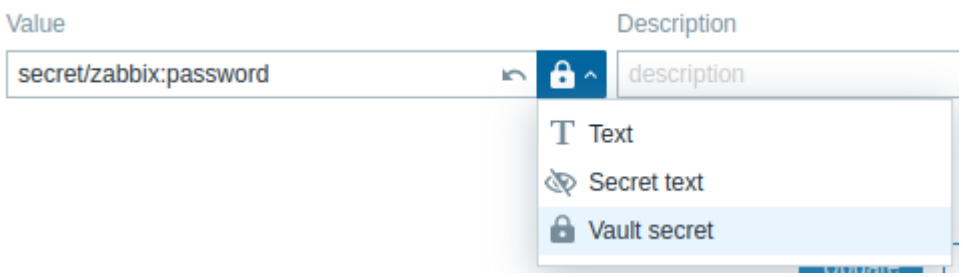
Po kliknięciu przycisku *Set new value* (lub zmianie typu wartości makra) bieżąca wartość zostanie usunięta. Możesz przywrócić oryginalną wartość, klikając strzałkę na końcu pola *Value* (dostępne tylko przed zapisaniem nowej konfiguracji). Pamiętaj, że przywrócenie oryginalnej wartości nie ujawni jej.

Note:
Adresy URL zawierające tajne makro nie będą działać, ponieważ makro w nich zostanie rozwiązane jako "******".

Sekret Vault

W przypadku makr sekretu Vault wartość makra jest przechowywana w zewnętrznym oprogramowaniu do zarządzania sekretami (vault).

Aby skonfigurować makro sekretu Vault, kliknij przycisk na końcu pola *Value* i wybierz opcję *Vault secret*:



Wartość makra musi wskazywać na sekret vault. Format danych wejściowych zależy od dostawcy vault. Przykłady konfiguracji dla poszczególnych dostawców znajdują się tutaj:

- HashiCorp
- CyberArk

Wartości makr sekretów Vault są pobierane z sejfu przez serwer Zabbix (oraz proxy Zabbix, jeśli opcja *Resolve secret vault macros by* jest ustawiona na *Zabbix server and proxy*) przy każdym odświeżeniu danych konfiguracyjnych, a następnie przechowywane w pamięci podręcznej konfiguracji. Serwer Zabbix i proxy Zabbix mogą używać różnych sejfów.

Jeśli opcja *Resolve secret vault macros by* jest ustawiona na *Zabbix server*, wówczas sekrety sejfu są pobierane tylko przez serwer, a proxy Zabbix otrzymuje wartości makr sekretów Vault z serwera Zabbix podczas każdej synchronizacji konfiguracji i przechowuje je we własnej pamięci podręcznej konfiguracji. Oznacza to, że proxy Zabbix nie może rozpocząć zbierania danych po ponownym uruchomieniu, dopóki nie otrzyma aktualizacji konfiguracji z serwera Zabbix.

Aby ręcznie odświeżyć wartości sekretów z sejfu, użyj opcji kontroli wykonania `secrets_reload` (tylko serwer).

Szyfrowanie musi być włączone między serwerem Zabbix a proxy; w przeciwnym razie rejestrowany jest komunikat ostrzegawczy serwera.

Warning:
Jeśli wartości makra nie można pomyślnie pobrać, odpowiednia pozycja używająca tej wartości stanie się nieobsługiwana.

Niezamaskowane lokalizacje

Ta lista zawiera lokalizacje parametrów, w których wartości makr tajnych są niezamaskowane.

Note:
Wartości makr tajnych pozostaną zamaskowane w poniższych lokalizacjach, jeśli będą odwoływane pośrednio. Na przykład {ITEM.KEY}, {ITEM.KEY<1-9>}, {LLDRULE.KEY} wbudowane makra używane w typach mediów (parametry Script lub Webhook) zostaną rozwiązane do kluczy pozycji zawierających zamaskowane tajne makra, takie jak `net.tcp.port[******,******]` zamiast `net.tcp.port[192.0.2.1,80]`.

Context		Parameter
Pozycje, prototypy pozycji, reguły LLD		
	Pozycja	*Parametry klucza pozycji*
	Prototyp pozycji	*Parametry klucza prototypu pozycji*
	Reguła wykrywania niskiego poziomu	*Parametry klucza pozycji wykrywania*
	agent SNMP	*Społeczność SNMP*
		Nazwa kontekstu (SNMPv3)
		Nazwa bezpieczeństwa (SNMPv3)
		Hasło uwierzytelniania (SNMPv3)
		Hasło prywatności (SNMPv3)
	agent HTTP	*URL*
		Pola zapytania
		Treść żądania
		Nagłówki
		Nazwa użytkownika
		Hasło
		Hasło klucza SSL
	Script	*Parametry*
		Skrypt
	Browser	*Parametry*
		Skrypt
	Monitor bazy danych	*Zapytanie SQL*
	agent TELNET	*Skrypt*
		Nazwa użytkownika
		Hasło
	agent SSH	*Skrypt*
		Nazwa użytkownika
		Hasło
	Simple check	*Nazwa użytkownika*
		Hasło
	agent JMX	*Nazwa użytkownika*
		Hasło

Context		Parameter
Wstępne przetwarzanie wartości pozycji		
	Krok wstępnego przetwarzania JavaScript	*Skrypt*
Scenariusze webowe		
	Scenariusz webowy	*Wartość zmiennej* *Wartość nagłówka* *URL* *Wartość pola zapytania* *Wartość pola POST* *Surowy POST*
	Uwierzytelnianie scenariusza webowego	*Użytkownik* *Hasło* *Hasło klucza SSL*
Łączniki		
	Łącznik	*URL* *Nazwa użytkownika* *Hasło* *Token* *Proxy HTTP* *Plik certyfikatu SSL* *Plik klucza SSL* *Hasło klucza SSL*
Wykrywanie sieci		
	SNMP	*Społeczność SNMP* *Nazwa kontekstu* (SNMPv3) *Nazwa bezpieczeństwa* (SNMPv3) *Hasło uwierzytelniania* (SNMPv3) *Hasło prywatności* (SNMPv3)
Skrypty globalne		
	Webhook	*Skrypt JavaScript* *Wartość parametru skryptu JavaScript*
	Telnet	*Nazwa użytkownika* *Hasło*
	SSH	*Nazwa użytkownika* *Hasło*
	Script	*Skrypt*
Typy mediów		
	Script	*Parametry skryptu*
	Webhook	*Parametry*
Zarządzanie IPMI		
	Host	*Nazwa użytkownika* *Hasło*

Makra niskopoziomowego wykrywania

Przegląd

Istnieje typ makra używanego w funkcji low-level discovery (LLD):

```
{#MACRO}
```

Jest to makro używane w regule LLD, które zwraca rzeczywiste wartości nazwy systemu plików, interfejsu sieciowego, SNMP OID itp.

Te makra mogą być używane do tworzenia prototypów encji (pozycja, wyzwalacz, wykres, reguła LLD, prototyp hosta i grupy hostów). Podczas wykrywania rzeczywistych systemów plików, interfejsów sieciowych, maszyn wirtualnych itp. makra te są zastępowane rzeczywistymi wartościami i stanowią podstawę do tworzenia rzeczywistych encji.

Niektóre makra low-level discovery są dostarczane „w pakiecie" z funkcją LLD w Zabbix — {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. Jednak podczas tworzenia niestandardowej reguły low-level discovery nie ma obowiązku stosowania tych nazw. Możesz wtedy użyć dowolnej innej nazwy makra LLD i odwoływać się do tej nazwy.

Obsługiwane typy danych

Podczas definiowania niestandardowych reguł wykrywania wartości właściwości zwracane w obiektach JSON dla makr LLD muszą być jednym z następujących typów prostych:

- string
- number
- boolean

Tablice, obiekty i wartości null nie są obsługiwane. Każde makro LLD odwołujące się do takiej wartości pozostanie nierozwinięte i będzie wyświetlane dosłownie (np. `'{#MY_MACRO}'`) podczas wstępnego przetwarzania i tworzenia pozycji.

Obsługiwane lokalizacje

Makra LLD mogą być używane:

- w filtrze reguły wykrywania niskiego poziomu
- w zagnieżdżonych regułach wykrywania niskiego poziomu, w
  - parametrach przetwarzania wstępnego JSONPath
  - polu JSONPath dla niestandardowych makr LLD
- dla prototypów pozycji i prototypów wykrywania w
  - nazwie
  - parametrach klucza
  - jednostce
  - interwale aktualizacji[1]
  - limicie czasu[1]
  - okresie przechowywania historii[1]
  - okresie przechowywania trendów[1]
  - krokach przetwarzania wstępnego wartości pozycji
  - OID SNMP
  - polu sensora IPMI
  - wyrażeniu pozycji obliczanej/agregowanej, w:
    - stałych wyrażenia i parametrach funkcji
    - parametrach klucza pozycji
  - warunkach filtra pozycji agregowanej (nazwa grupy hostów i nazwa tagu)
  - skrypcie SSH i skrypcie Telnet
  - zapytaniu SQL monitorowania bazy danych
  - polu endpointu pozycji JMX
  - opisie
  - polu URL agenta HTTP
  - polu pól zapytania HTTP agenta HTTP
  - polu treści żądania agenta HTTP
  - polu wymaganych kodów statusu agenta HTTP
  - kluczu i wartości pola nagłówków agenta HTTP
  - polu nazwy użytkownika uwierzytelniania HTTP agenta HTTP
  - polu hasła uwierzytelniania HTTP agenta HTTP
  - polu proxy HTTP agenta HTTP
  - polu pliku certyfikatu SSL HTTP agenta HTTP
  - polu pliku klucza SSL HTTP agenta HTTP
  - polu hasła klucza SSL HTTP agenta HTTP
  - tagach
- dla prototypów wyzwalaczy w
  - nazwie
  - danych operacyjnych
  - wyrażeniu (tylko w stałych i parametrach funkcji)
  - URL
  - opisie
  - tagach
- dla prototypów wykresów w
  - nazwie
- dla prototypów hostów w
  - nazwie
  - widocznej nazwie
  - niestandardowych polach interfejsu: IP, DNS, port, społeczność SNMP v1/v2, nazwa kontekstu SNMP v3, nazwa zabezpieczeń SNMP v3, hasło uwierzytelniania SNMP v3, hasło prywatności SNMP v3
  - nazwie prototypu grupy hostów
  - wartości tagu hosta

- wartości makra hosta
- (zobacz pełną listę)

We wszystkich tych miejscach, z wyjątkiem filtra reguły wykrywania niskiego poziomu, makra LLD mogą być używane wewnątrz statycznego kontekstu makra użytkownika.

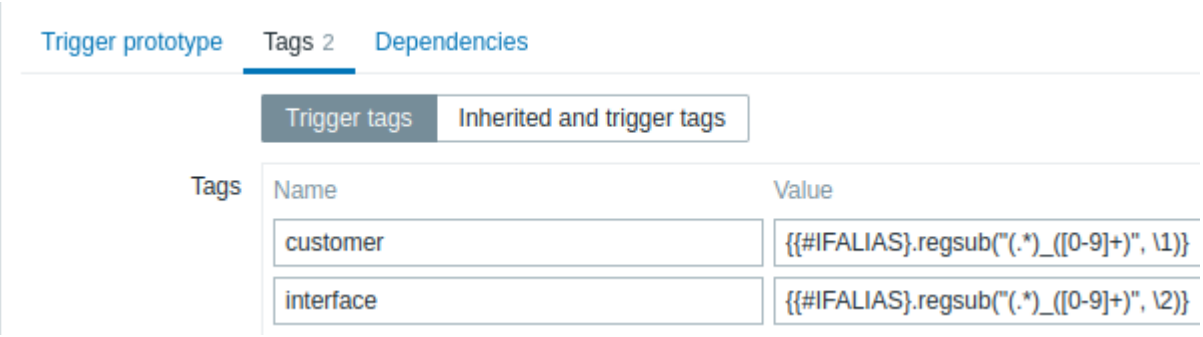
Korzystanie z funkcji makr

Funkcje makr są obsługiwane w makrach niskopoziomowego wykrywania (z wyjątkiem filtra reguły niskopoziomowego wykrywania), co umożliwia wyodrębnienie określonej części wartości makra za pomocą wyrażenia regularnego.

Na przykład można chcieć wyodrębnić nazwę klienta i numer interfejsu z następującego makra LLD na potrzeby tagowania zdarzeń:

```
{#IFALIAS}=customername_1
```

W tym celu można użyć funkcji makra `regsub` wraz z makrem w polu wartości tagu zdarzenia prototypu wyzwalacza:



Należy pamiętać, że przecinki nie są dozwolone w nieujętych w cudzysłowy parametrach klucza pozycji, dlatego parametr zawierający funkcję makra musi być ujęty w cudzysłowy. Do oznaczania znaków cudzysłowu wewnątrz parametru należy użyć znaku odwrotnego ukośnika (\). Przykład:

```
net.if.in["{{#IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}",bytes]
```

Więcej informacji na temat składni funkcji makr można znaleźć tutaj: Funkcje makr

Funkcje makr są obsługiwane w makrach niskopoziomowego wykrywania od wersji Zabbix 4.0.

Prototypy pozycji bez parametrów klucza

Gdy parametry klucza pozycji nie są używane, umieść makro LLD wewnątrz nawiasów parametrów klucza pozycji `[...]`, aby było traktowane jako parametr i zastępowane podczas wykrywania:

```
v_[{#MACRO}]
```

Przypisy

1 W polach oznaczonych [1] pojedyncze makro musi wypełniać całe pole. Wiele makr w polu lub makra zmieszane z tekstem nie są obsługiwane.

Makra wyrażeń

Omówienie

Makra wyrażeń umożliwiają wykonywanie obliczeń w polach.

Ich wartość jest obliczana przez najpierw rozwiązanie wszystkich zagnieżdżonych makr, a następnie ocenę wynikowego wyrażenia.

Składnia:

```
{?EXPRESSION}
```

`EXPRESSION` używa tej samej składni i obsługuje te same funkcje co wyrażenia wyzwalaczy.

Przykład:

```
{?trendavg(/host/item1,1M:now/M)/trendavg(/host/item1,1M:now/M-1y)*100}
```

W miejscach, które obsługują makra wyrażeń, wyszukaj "{?EXPRESSION}" w tabeli obsługiwanych makr.

Uwagi dotyczące użycia:

- Używaj makr {FUNCTION.*}, aby odwoływać się do wartości funkcji wyrażeń wyzwalaczy/wyrażeń odzyskiwania.
- Używaj makr {HOST.HOST<1-9>} i {ITEM.KEY<1-9>}, aby odwoływać się do hostów i pozycji.

- W szablonach używaj makr {HOST.HOST<1-9>} albo pomiń hosta całkowicie dla pierwszego hosta — na przykład {?avg(//item1,1h)} — zamiast nazw szablonów, ponieważ nazwy szablonów nie są zastępowane nazwami hostów podczas łączenia szablonów.

```
{?{FUNCTION.VALUE2} - {FUNCTION.VALUE3}}
{?max(/{HOST.HOST}/{ITEM.KEY},3h)}
```

Zobacz też przykłady wyrażeń wyzwalaczy, aby zobaczyć przykład użycia makra wyrażenia w nazwie zdarzenia.

Użytkownicy i grupy użytkowników

Przegląd

Wszyscy użytkownicy w Zabbix uzyskują dostęp do aplikacji Zabbix za pośrednictwem internetowego frontend. Każdemu użytkownikowi przypisywana jest unikalna nazwa logowania i hasło.

Wszystkie hasła użytkowników są szyfrowane i przechowywane w bazie danych Zabbix. Użytkownicy nie mogą używać swojego identyfikatora użytkownika i hasła do bezpośredniego logowania się do serwera UNIX, chyba że zostali również odpowiednio skonfigurowani w UNIX. Komunikacja między serwerem WWW a przeglądarką użytkownika może być chroniona przy użyciu SSL.

Dzięki elastycznemu schematowi uprawnień użytkowników można ograniczać i różnicować prawa do:

- Dostępu do administracyjnych funkcji frontend Zabbix.
- Wykonywania określonych działań w frontend.
- Dostępu do monitorowanych hostów w grupach hostów.
- Korzystania z określonych metod API.

Konfigurowanie użytkownika

Przegląd

Początkowa instalacja Zabbix zawiera dwóch predefiniowanych użytkowników:

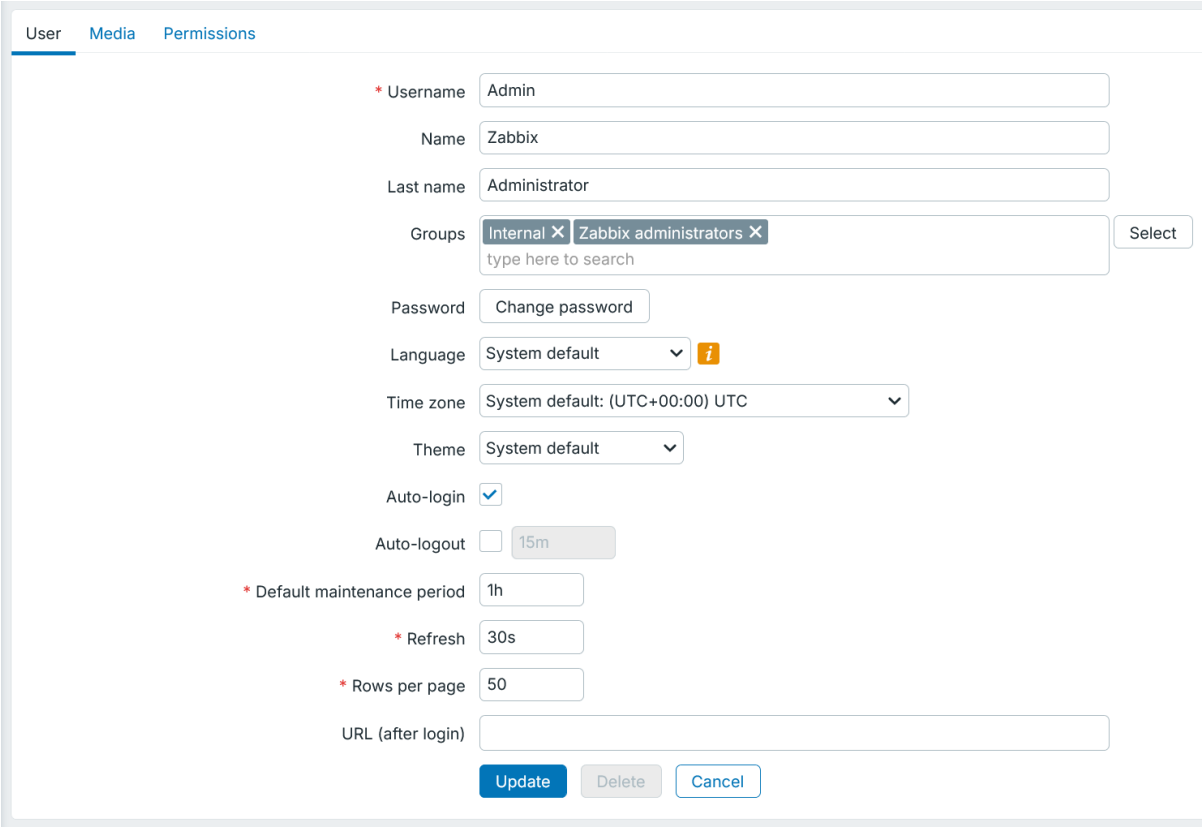
- *Admin* - superużytkownik Zabbix z pełnymi uprawnieniami.
- *guest* - specjalny użytkownik Zabbix. Użytkownik „guest” jest domyślnie wyłączony. Aby włączyć użytkownika, usuń predefiniowaną grupę *Disabled* z grup użytkownika. Możesz zalogować się za pomocą tego użytkownika i uzyskać dostęp do stron monitorowania w Zabbix. Pamiętaj, że domyślnie użytkownik „guest” nie ma uprawnień do obiektów Zabbix.

Aby skonfigurować użytkownika:

1. Przejdź do *Users* > *Users*.
2. Kliknij *Create user* (lub nazwę użytkownika, aby edytować istniejącego użytkownika).
3. Edytuj atrybuty użytkownika w formularzu.

Atrybuty ogólne

Karta *Użytkownik* zawiera ogólne atrybuty użytkownika:



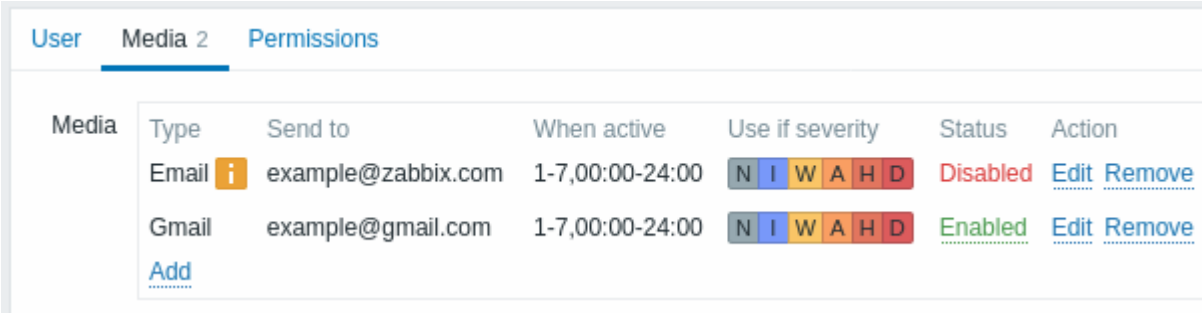
Wszystkie wymagane pola są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa użytkownika	Unikalna nazwa użytkownika używana jako nazwa logowania.
Imię	Imię użytkownika (opcjonalne). Jeśli nie jest puste, jest widoczne w informacjach o potwierdzeniu i odbiorcy powiadomienia.
Nazwisko	Nazwisko użytkownika (opcjonalne). Jeśli nie jest puste, jest widoczne w informacjach o potwierdzeniu i odbiorcy powiadomienia.
Grupy	Wybierz grupy użytkowników, do których należy użytkownik. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy użytkowników wyświetli listę rozwijaną pasujących grup. Przewiń w dół, aby wybrać grupę. Możesz też kliknąć *Wybierz*, aby dodać grupy. Kliknij „x", aby usunąć wybraną grupę. Przynależność do grup użytkowników określa, do jakich grup hostów i hostów użytkownik ma dostęp.
Hasło	Dwa pola do wprowadzenia hasła użytkownika lub przycisk *Zmień hasło*, jeśli użytkownik już istnieje. Kliknięcie przycisku *Zmień hasło* otwiera dwa pola do wprowadzenia nowego hasła. W przypadku użytkownika z *rolą superadministratora*, który zmienia własne hasło, kliknięcie przycisku *Zmień hasło* otwiera dodatkowe pole do wprowadzenia bieżącego (starego) hasła. Po pomyślnej zmianie hasła użytkownik, którego hasło zostało zmienione, zostanie wylogowany ze wszystkich aktywnych sesji. Należy pamiętać, że hasło można zmienić tylko użytkownikom korzystającym z uwierzytelniania wewnętrznego Zabbix.
Język	Język frontendu Zabbix. Do działania tłumaczeń wymagane jest rozszerzenie php gettext.
Strefa czasowa	Wybierz strefę czasową, aby zastąpić globalną strefę czasową na poziomie użytkownika, lub wybierz **Domyślna systemowa**, aby używać globalnych ustawień strefy czasowej.
Motyw	Określa wygląd frontendu: **Domyślny systemowy** - używa domyślnych ustawień systemowych **Niebieski** - standardowy niebieski motyw **Niebieski (klasyczny)** - standardowy niebieski motyw z użyciem starszych czcionek **Ciemny** - alternatywny ciemny motyw **Ciemnoniebieski** - ciemnoniebieski motyw **Ciemny (klasyczny)** - alternatywny ciemny motyw z użyciem starszych czcionek **Jasny o wysokim kontraście** - jasny motyw o wysokim kontraście **Ciemny o wysokim kontraście** - ciemny motyw o wysokim kontraście

Parametr	Opis
Automatyczne logowanie	Zaznacz to pole wyboru, aby Zabbix zapamiętał użytkownika i automatycznie logował go przez 30 dni. Podczas logowania z opcją *Zapamiętaj przez 30 dni*: - Automatyczne wylogowanie jest resetowane (sesja trwa 30 dni). - Automatyczne logowanie jest włączone na potrzeby ponownego uwierzytelniania bez dodatkowych działań. Podczas logowania bez opcji *Zapamiętaj przez 30 dni*: - Automatyczne logowanie jest włączone na potrzeby ponownego uwierzytelniania bez dodatkowych działań. - Automatyczne wylogowanie nadal podlega standardowemu ustawieniu limitu czasu. Do tego celu używane są pliki cookie przeglądarki.
Automatyczne wylogowanie	Po zaznaczeniu tego pola wyboru użytkownik zostanie automatycznie wylogowany po upływie ustawionej liczby sekund (minimum 90 sekund, maksimum 1 dzień). Należy pamiętać, że to ustawienie jest zastępowane, jeśli opcja *Zapamiętaj przez 30 dni* jest włączona, ponieważ sesja zostaje przedłużona na cały ten okres. Obsługiwane są przyrostki czasu, np. 90s, 5m, 2h, 1d. Należy pamiętać, że ta opcja nie będzie działać: * Jeśli globalna opcja konfiguracji „Pokaż ostrzeżenie, jeśli serwer Zabbix nie działa” jest włączona, a frontend Zabbix pozostaje otwarty. * Gdy strony menu Monitorowanie wykonują odświeżanie informacji w tle. * Jeśli logowanie odbywa się z zaznaczoną opcją *Zapamiętaj mnie przez 30 dni*.
Domyślny okres konserwacji	Ustaw domyślny okres konserwacji dla konserwacji ad hoc.
Odświeżanie	Ustaw częstotliwość odświeżania używaną dla wykresów, danych w postaci zwykłego tekstu itp. Można ustawić wartość 0, aby wyłączyć odświeżanie. Obsługiwane są przyrostki czasu, np. 90s, 5m, 1h.
Wierszy na stronę	Możesz określić, ile wierszy na stronę będzie wyświetlanych na listach.
URL (po zalogowaniu)	Możesz skonfigurować Zabbix tak, aby po pomyślnym zalogowaniu przekierowywał użytkownika do określonego adresu URL, na przykład do strony *Problemy*.

Media użytkownika

Karta *Media* zawiera listę wszystkich mediów zdefiniowanych dla użytkownika. Media są używane do wysyłania powiadomień.



Kliknij *Dodaj*, aby przypisać media do użytkownika.

Jeśli typ mediów został wyłączony:

- Po nazwie wyświetlana jest żółta ikona informacji.
- W kolumnie Status wyświetlana jest wartość *Wyłączone*.

Note:
Uprawnienia użytkownika do zmiany szczegółów własnych mediów można nadawać i odbierać na podstawie jego roli użytkownika (zobacz opcję *Tworzenie i edytowanie własnych mediów*). Uprawnienia użytkownika z uprawnieniami super administratora do zmiany szczegółów mediów innych użytkowników również można nadawać i odbierać na podstawie jego roli użytkownika (zobacz opcję *Tworzenie i edytowanie mediów użytkowników*).

Więcej informacji na temat konfigurowania mediów użytkownika można znaleźć w sekcji Typy mediów.

Uprawnienia

Karta *Uprawnienia* zawiera informacje o następujących elementach:

- Rola użytkownika (wymagana dla każdego nowo utworzonego użytkownika), którą może zmienić wyłącznie użytkownik typu *Super admin*.

Attention:
Użytkowników nie można tworzyć bez roli użytkownika (z wyjątkiem korzystania z User API Zabbix). Wcześniej utworzeni użytkownicy, którzy nie mają przypisanej roli, mogą nadal być edytowani bez przypisywania im roli. Jednak po przypisaniu roli można ją tylko zmienić, a nie usunąć.

 Należy pamiętać, że użytkownicy bez roli mogą logować się do Zabbix wyłącznie przy użyciu uwierzytelniania LDAP lub SAML, pod warunkiem że ich informacje LDAP/SAML odpowiadają mapowaniom grup użytkowników skonfigurowanym w Zabbix.

- Typ użytkownika (*User*, *Admin*, *Super admin*) określony w konfiguracji roli użytkownika.
- Grupy hostów i szablonów, do których użytkownik ma dostęp.
  - Użytkownicy typu *User* i *Admin* domyślnie nie mają dostępu do żadnych grup, szablonów ani hostów. Aby przyznać taki dostęp, użytkownicy muszą zostać uwzględnieni w grupach użytkowników skonfigurowanych z uprawnieniami do odpowiednich elementów.
- Prawa dostępu do sekcji i elementów frontend Zabbix, modułów oraz metod API.
  - Elementy, do których dostęp jest dozwolony, są wyświetlane na zielono, natomiast elementy, do których dostęp jest zabroniony, są wyświetlane na jasnoszaro.
  - Jeśli dostęp do API jest włączony, metody API roli użytkownika są wyświetlane w sekcji *Allowed methods* lub *Denied methods*, zależnie od tego, czy w konfiguracji roli użytkownika wybrano *Allow list* czy *Deny list*. Jeśli nie określono żadnych metod API, wyświetlana jest wartość *None*. Wartość *None* w sekcji *Allowed methods* oznacza, że żadna metoda API nie jest dozwolona, natomiast *None* w sekcji *Denied methods* oznacza, że dozwolone są wszystkie metody API dostępne dla danego typu użytkownika.
- Uprawnienia do wykonywania określonych działań.
  - Działania, które użytkownik może wykonywać, są wyświetlane na zielono, natomiast działania, których wykonywanie jest zabronione, są wyświetlane na jasnoszaro.

Szczegółowe informacje można znaleźć na stronie Uprawnienia.

Uprawnienia

Przegląd

Uprawnienia w Zabbix zależą od typu użytkownika, dostosowanych ról użytkowników oraz dostępu do hostów, który jest określany na podstawie grupy użytkowników.

Typy użytkowników

Uprawnienia w Zabbix zależą przede wszystkim od typu użytkownika:

- *Użytkownik* - ma ograniczone prawa dostępu do sekcji menu (zobacz poniżej) i domyślnie nie ma dostępu do żadnych zasobów. Wszelkie uprawnienia do grup hostów lub szablonów muszą zostać jawnie przypisane.
- *Administrator* - ma niepełne prawa dostępu do sekcji menu (zobacz poniżej). Domyślnie użytkownik nie ma dostępu do żadnych grup hostów. Wszelkie uprawnienia do grup hostów lub szablonów muszą zostać jawnie nadane.
- *Superadministrator* - ma dostęp do wszystkich sekcji menu. Użytkownik ma dostęp do odczytu i zapisu do wszystkich grup hostów i szablonów. Uprawnień nie można odebrać przez odmowę dostępu do określonych grup.

Dostęp do menu

Note:
Ograniczenie dostępu do niektórych elementów interfejsu użytkownika uniemożliwia jedynie otwarcie danej strony — nie usuwa możliwości dostępu do danych bazowych w innych częściach interfejsu.

Poniższa tabela przedstawia dostęp do sekcji menu Zabbix w zależności od typu użytkownika:

Sekcja menu		Użytkownik	Administrator	Superadministrator
Pulpity		+	+	+
Monitorowanie		+	+	+
	Problemy	+	+	+
	Hosty	+	+	+
	Najnowsze dane	+	+	+
	Mapy	+	+	+
	Wykrywanie		+	+
APM		+	+	+

Sekcja menu		Użytkownik	Administrator	Superadministrator
	Ślady	+	+	+
	Metryki	+	+	+
	Dzienniki	+	+	+
Usługi		+	+	+
	Usługi	+	+	+
	SLA		+	+
	Raport SLA	+	+	+
Inwentaryzacja		+	+	+
	Przegląd	+	+	+
	Hosty	+	+	+
Raporty		+	+	+
	Informacje o systemie		+	+
	Raporty zaplanowane		+	+
	Raport dostępności	+	+	+
	100 najważniejszych wyzwalaczy	+	+	+
	Dziennik audytu			+
	Dziennik działań			+
	Powiadomienia		+	+
Gromadzenie danych			+	+
	Grupy szablonów		+	+
	Grupy hostów		+	+
	Szablony		+	+
	Hosty		+	+
	Okresy utrzymania		+	+
	Korelacja zdarzeń			+
	Wykrywanie		+	+
Alerty			+	+
	Działania wyzwalaczy		+	+
	Działania usług		+	+
	Działania wykrywania		+	+
	Działania automatycznej rejestracji		+	+
	Działania wewnętrzne		+	+
	Typy mediów			+
	Skrypty			+
Użytkownicy				+
	Grupy użytkowników			+
	Role użytkowników			+
	Użytkownicy			+
	Tokeny API			+
	Uwierzytelnianie			+
	Urządzenia			+
Administracja				+
	Źródło danych			+
	Ogólne			+
	Dziennik audytu			+
	Porządkowanie			+
	Grupy proxy			+
	Proxy			+
	Makra			+
	Kolejka			+

Role użytkowników

Role użytkowników umożliwiają niestandardowe dostosowanie uprawnień zdefiniowanych przez typ użytkownika. Nie można dodawać uprawnień (ponieważ przekraczałoby to uprawnienia typu użytkownika), ale niektóre uprawnienia można odebrać.

Ponadto rola użytkownika określa dostęp nie tylko do sekcji menu, ale także do usług, modułów, metod API i różnych działań w frontend.

Role użytkowników są konfigurowane przez użytkowników Super admin w sekcji *Użytkownicy* > *Role użytkowników*.

Role użytkowników są przypisywane użytkownikom przez użytkowników Super admin w formularzu konfiguracji użytkownika, na karcie *Uprawnienia*.

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

Dostęp do hostów

Dostęp do danych dowolnego hosta i szablonu w Zabbix jest przyznawany grupom użytkowników wyłącznie na poziomie grup hostów/szablonów.

Oznacza to, że pojedynczemu użytkownikowi nie można bezpośrednio przyznać dostępu do hosta (ani grupy hostów). Dostęp do hosta można przyznać tylko poprzez członkostwo w grupie użytkowników, której przyznano dostęp do grupy hostów zawierającej tego hosta.

Podobnie, użytkownik może uzyskać dostęp do szablonu wyłącznie poprzez członkostwo w grupie użytkowników, której przyznano dostęp do grupy szablonów zawierającej dany szablon.

Grupy użytkowników

Przegląd

Grupy użytkowników umożliwiają grupowanie użytkowników zarówno do celów organizacyjnych, jak i w celu przypisywania uprawnień do danych. Uprawnienia do wyświetlania i konfigurowania danych grup hostów i grup szablonów są przypisywane grupom użytkowników, a nie poszczególnym użytkownikom.

Często zasadne może być rozdzielenie informacji dostępnych dla jednej grupy użytkowników od informacji dostępnych dla innej grupy. Można to osiągnąć, grupując użytkowników, a następnie przypisując różne uprawnienia do grup hostów i grup szablonów.

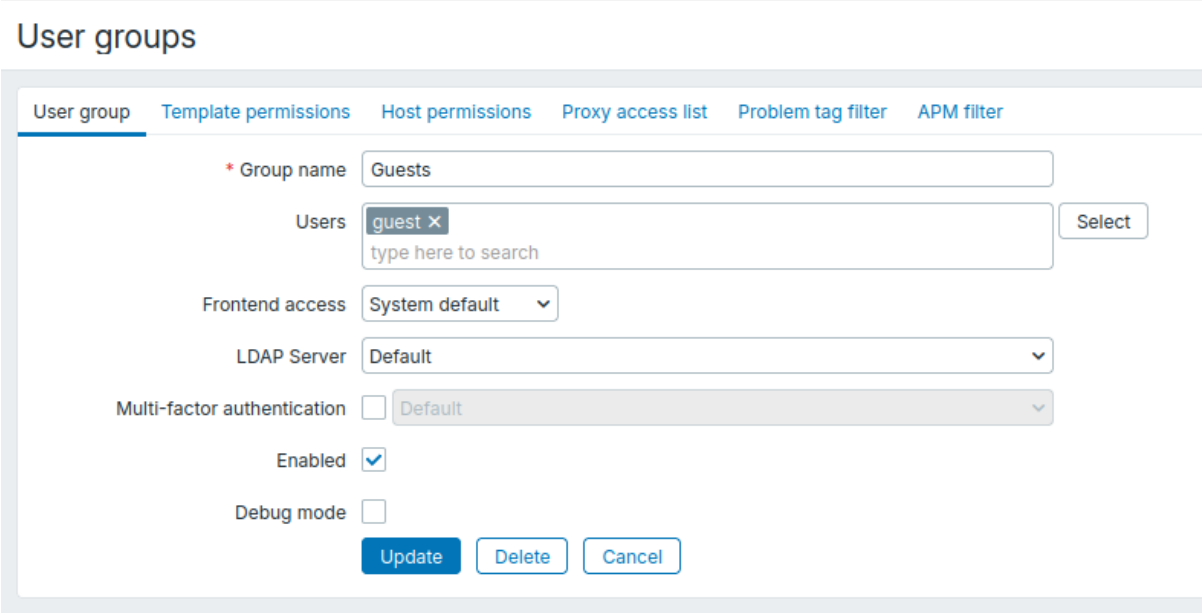
Użytkownik może należeć do dowolnej liczby grup.

Konfiguracja

Aby skonfigurować grupę użytkowników:

1. Przejdź do *Użytkownicy* > *Grupy użytkowników*.
2. Kliknij *Utwórz grupę użytkowników* (lub nazwę grupy, aby edytować istniejącą grupę).
3. Edytuj atrybuty grupy w formularzu.

Karta **Grupa użytkowników** zawiera ogólne atrybuty grupy:

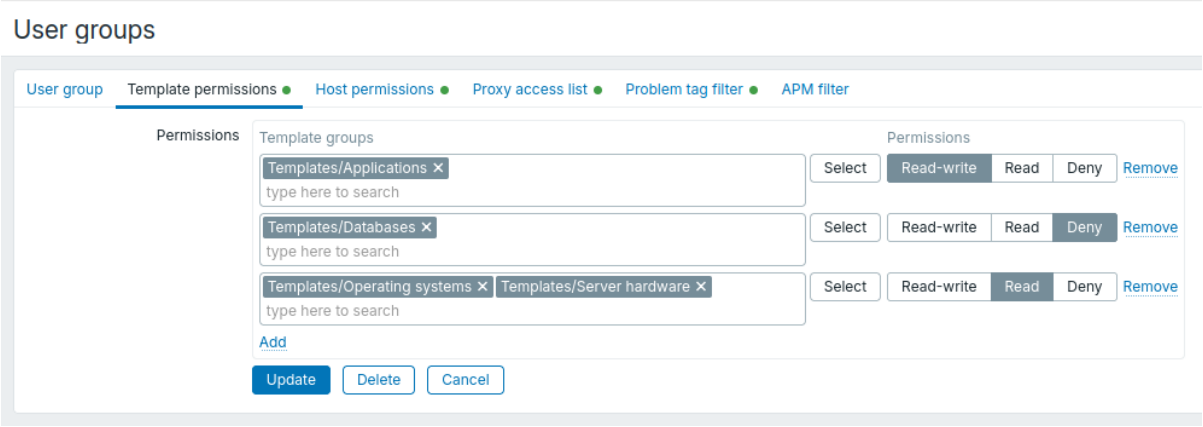


Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

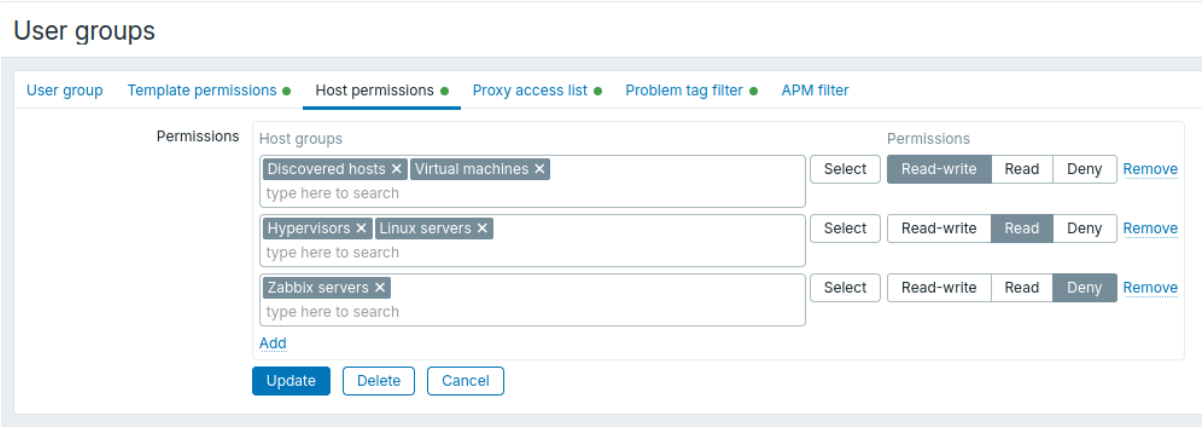
Parametr	Opis
Nazwa grupy	Unikalna nazwa grupy.
Użytkownicy	Aby dodać użytkowników do grupy, zacznij wpisywać nazwę istniejącego użytkownika. Gdy pojawi się lista rozwijana z pasującymi nazwami użytkowników, przewiń ją w dół, aby wybrać użytkownika. Możesz również kliknąć przycisk *Wybierz*, aby wybrać użytkowników w wyskakującym oknie.

Parametr	Opis
Dostęp do frontendu	Sposób uwierzytelniania użytkowników należących do grupy. **Domyślne ustawienia systemowe** - używa domyślnej metody uwierzytelniania (ustawionej globalnie) **Wewnętrzne** - używa wewnętrznego uwierzytelniania Zabbix (nawet jeśli globalnie używane jest uwierzytelnianie LDAP). Ignorowane, jeśli domyślnie globalnie używane jest uwierzytelnianie HTTP. **LDAP** - używa uwierzytelniania LDAP (nawet jeśli globalnie używane jest uwierzytelnianie wewnętrzne). Ignorowane, jeśli domyślnie globalnie używane jest uwierzytelnianie HTTP. **Wyłączone** - dostęp do frontendu Zabbix jest zabroniony dla tej grupy
Serwer LDAP	Wybierz, który serwer LDAP ma być używany do uwierzytelniania użytkownika. To pole jest włączone tylko wtedy, gdy dla opcji *Dostęp do frontendu* wybrano LDAP lub Domyślne ustawienia systemowe.
Uwierzytelnianie wieloskładnikowe	Wybierz, która metoda uwierzytelniania wieloskładnikowego ma być używana do uwierzytelniania użytkownika: **Domyślna** - używa metody ustawionej jako domyślna w konfiguracji MFA; ta opcja jest domyślnie wybrana dla nowych grup użytkowników, jeśli MFA jest włączone; **<Nazwa metody>** - używa wybranej metody (na przykład „Zabbix TOTP"); **Wyłączone** - MFA jest wyłączone dla tej grupy; ta opcja jest domyślnie wybrana dla nowych grup użytkowników, jeśli MFA jest wyłączone. Należy pamiętać, że jeśli użytkownik należy do wielu grup użytkowników z włączonym MFA (lub co najmniej jedna grupa ma włączone MFA), obowiązują następujące zasady uwierzytelniania: jeśli dowolna grupa używa metody MFA „Domyślna", zostanie ona użyta do uwierzytelnienia użytkownika; w przeciwnym razie do uwierzytelniania zostanie użyta pierwsza metoda (uporządkowana alfabetycznie).
Włączone	Stan grupy użytkowników i jej członków. *Zaznaczone* - grupa użytkowników i użytkownicy są włączeni *Niezaznaczone* - grupa użytkowników i użytkownicy są wyłączeni
Tryb debugowania	Zaznacz to pole wyboru, aby aktywować tryb debugowania dla użytkowników.

Karta **Uprawnienia do szablonów** umożliwia określenie dostępu grupy użytkowników do grup szablonów, a za ich pośrednictwem do danych szablonów:



Karta **Uprawnienia do hostów** umożliwia określenie dostępu grupy użytkowników do grup hostów, a za ich pośrednictwem do danych hostów:



Kliknij Add, aby wybrać grupy szablonów/hostów (nadrzędne lub zagnieżdżone) i przypisać im uprawnienia. Zacznij wpisywać nazwę grupy (pojawi się lista rozwijana z pasującymi grupami) lub kliknij *Wybierz*, aby otworzyć wyskakujące okno zawierające listę wszystkich grup.

Następnie użyj przycisków opcji, aby przypisać uprawnienia wybranym grupom. Dostępne są następujące uprawnienia:

- **Odczyt i zapis** - dostęp do grupy z uprawnieniami odczytu i zapisu
- **Odczyt** - dostęp do grupy tylko do odczytu
- **Odmów** - odmowa dostępu do grupy

Jeśli ta sama grupa szablonów/hostów zostanie dodana w kilku wierszach z różnymi ustawieniami uprawnień, zastosowane zostanie najbardziej restrykcyjne uprawnienie.

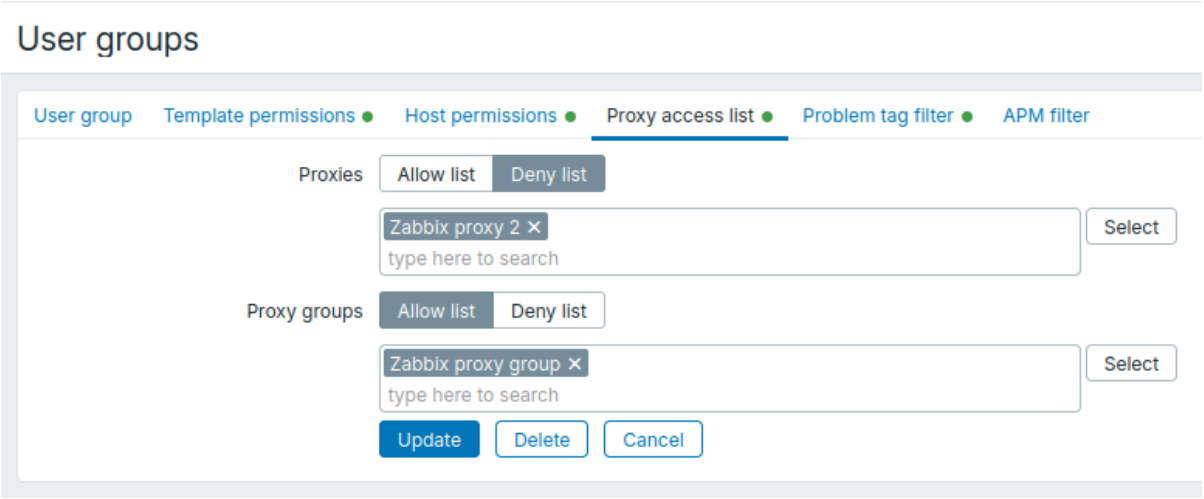
Należy pamiętać, że użytkownik *Super admin* może wymusić, aby zagnieżdżone grupy miały taki sam poziom uprawnień jak grupa nadrzędna; można to zrobić w formularzu konfiguracji grup hostów/szablonów.

Karty **Uprawnienia do szablonów** i **Uprawnienia do hostów** obsługują ten sam zestaw parametrów.

Bieżące uprawnienia do grup są wyświetlane w bloku *Uprawnienia* i można je modyfikować lub usuwać.

Note:
Jeśli grupa użytkowników ma uprawnienia **Odczyt i zapis** do hosta oraz uprawnienie **Odmów** lub nie ma uprawnień do szablonu połączonego z tym hostem, użytkownicy należący do tej grupy nie będą mogli edytować pozycji utworzonych na podstawie szablonu na hoście, a nazwa szablonu będzie wyświetlana jako *Niedostępny szablon*.

Karta **Lista dostępu do proxy** umożliwia określenie dostępu grupy użytkowników do proxy, indywidualnie lub za pośrednictwem grup proxy:



Użyj pola *Grupy proxy*, aby zastosować uprawnienie do wybranych grup. Proxy należące do grupy można zezwolić lub zablokować wyłącznie za pośrednictwem tej grupy. Użyj pola *Proxy*, aby dodać pojedyncze proxy, które nie należą do żadnej grupy proxy.

Pola *Proxy* i *Grupy proxy* obsługują ten sam zestaw parametrów.

Kliknij przycisk *Wybierz* obok pola *Proxy* lub *Grupy proxy*, aby odpowiednio wybrać proxy lub grupy proxy. Możesz również zacząć wpisywać nazwę proxy lub grupy proxy w odpowiednim polu. Pojawi się lista rozwijana pasujących proxy lub grup proxy.

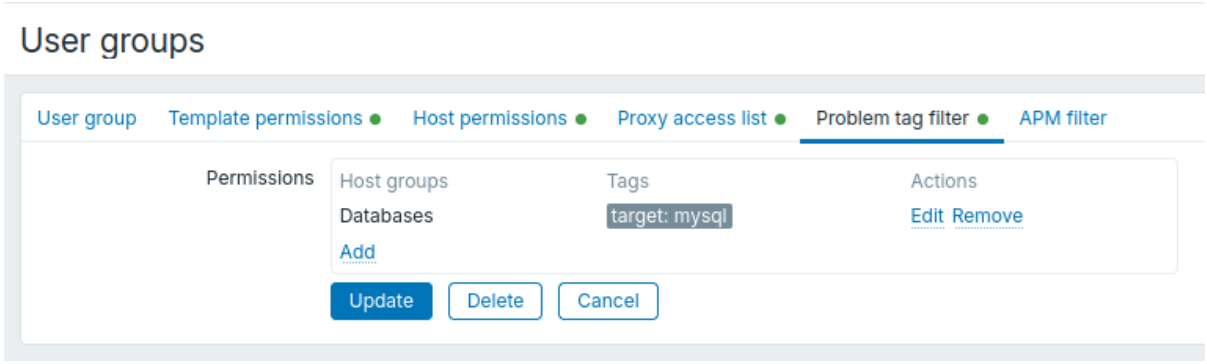
Wybierz *Lista dozwolonych*, aby zezwolić grupie użytkowników na dostęp wyłącznie do wybranych proxy i grup proxy.
 Jeśli wybór *Proxy* dla opcji *Lista dozwolonych* jest pusty, grupa użytkowników nie ma dostępu do żadnego proxy, które nie należy do grupy proxy. Jeśli wybór *Grupy proxy* dla opcji *Lista dozwolonych* jest pusty, grupa użytkowników nie ma dostępu do żadnej grupy proxy.

Wybierz *Lista zablokowanych*, aby odmówić grupie użytkowników dostępu do wybranych proxy i grup proxy. Dostęp do wszystkich pozostałych proxy i grup proxy jest dozwolony.
 Jeśli wybór *Proxy* dla opcji *Lista zablokowanych* jest pusty, grupa użytkowników ma pełny dostęp do wszystkich proxy, które nie należą do grupy proxy. Jeśli wybór *Grupy proxy* dla opcji *Lista zablokowanych* jest pusty, grupa użytkowników ma pełny dostęp do wszystkich grup proxy.

Zasady pierwszeństwa uprawnień dla użytkowników należących do wielu grup użytkowników:

- *Lista zablokowanych* ma pierwszeństwo przed *Listą dozwolonych*. Jeśli dowolna grupa odmówi dostępu do proxy/grupy proxy, użytkownik nie będzie mieć do niego dostępu.
- Uprawnienia *Listy dozwolonych* są kumulatywne. Jeśli wiele grup zezwala na dostęp, użytkownik może uzyskać dostęp do wszystkich proxy dozwolonych przez te grupy.

Karta **Filtr tagów problemów** umożliwia ustawienie uprawnień opartych na tagach dla grup użytkowników, aby wyświetlać problemy filtrowane według nazwy i wartości tagu:

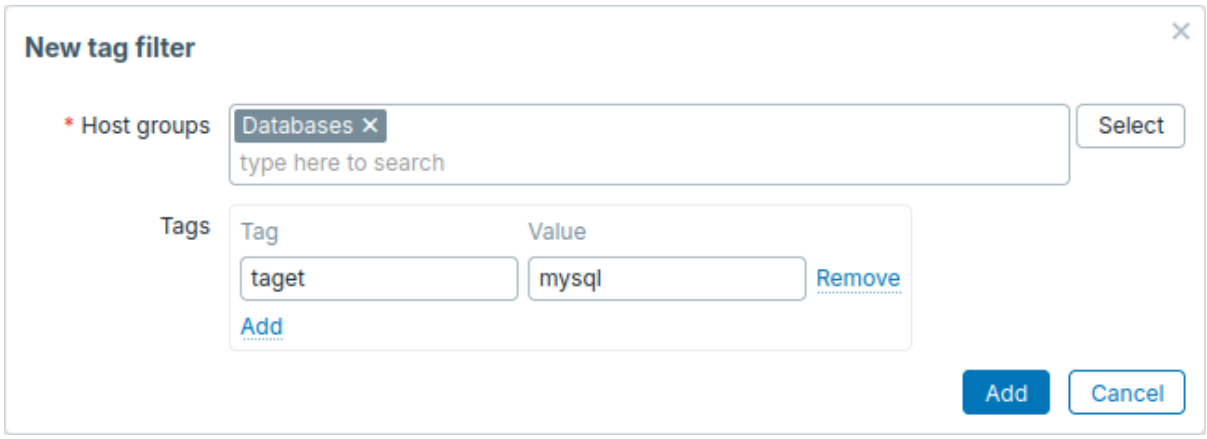


Kliknij Add, aby wybrać grupy hostów. Aby wybrać grupę hostów, dla której ma zostać zastosowany filtr tagów, kliknij *Wybierz*, aby uzyskać pełną listę istniejących grup hostów, lub zacznij wpisywać nazwę grupy hostów, aby wyświetlić listę rozwijaną pasujących grup. Wyświetlane będą tylko grupy hostów, ponieważ filtra tagów problemów nie można zastosować do grup szablonów.

Można dodawać nazwy tagów bez wartości, ale nie można dodawać wartości bez nazw. W bloku *Uprawnienia* wyświetlane są tylko trzy pierwsze tagi (wraz z wartościami, jeśli je mają); jeśli jest ich więcej, można je wyświetlić, klikając ikonę ••• lub umieszczając na niej wskaźnik myszy.

Filtr tagów umożliwia oddzielenie dostępu do grupy hostów od możliwości wyświetlania problemów.

Na przykład jeśli administrator bazy danych musi widzieć tylko problemy dotyczące baz danych „MySQL", najpierw należy utworzyć grupę użytkowników dla administratorów baz danych, a następnie określić nazwę tagu „target" i wartość „mysql".



Jeśli określono nazwę tagu „target", a pole wartości pozostawiono puste, grupa użytkowników będzie widzieć wszystkie problemy z tagiem o nazwie „target" dla wybranej grupy hostów.

Upewnij się, że nazwa tagu i jego wartość zostały określone poprawnie; w przeciwnym razie grupa użytkowników nie będzie widzieć żadnych problemów.

Przeanalizujmy przykład, w którym użytkownik jest członkiem kilku wybranych grup użytkowników. Filtrowanie w tym przypadku będzie używać warunku OR dla tagów.

Grupa użytkowników A			**Grupa użytkowników B**			**Widoczny wynik dla użytkownika (członka) obu grup**
Filtr tagów						
Grupa hostów	*Nazwa tagu*	*Wartość tagu*	*Grupa hostów*	*Nazwa tagu*	*Wartość tagu*	
Bazy danych	target	mysql	Bazy danych	target	oracle	problemy target:mysql lub target:oracle są widoczne
Nie skonfigurowano w **Filtrze tagów problemów**			Bazy danych	target	oracle	problemy target:oracle są widoczne

Dostęp z kilku grup użytkowników

Użytkownik może należeć do dowolnej liczby grup użytkowników. Grupy te mogą mieć różne uprawnienia dostępu do hostów lub szablonów.

Dlatego ważne jest, aby wiedzieć, do jakich encji nieuprzywilejowany użytkownik będzie miał w rezultacie dostęp. W poniższym przykładzie rozważmy, jak dostęp do hosta **X** (w grupie hostów 1) będzie zależał od różnych sytuacji dla użytkownika należącego do grup użytkowników A i B.

- Jeśli grupa A ma tylko dostęp *Odczyt* do grupy hostów 1, a grupa B ma dostęp *Odczyt-zapis* do grupy hostów 1, użytkownik otrzyma dostęp **Odczyt-zapis** do „X”.

Attention:
Uprawnienia „Odczyt-zapis” mają pierwszeństwo przed uprawnieniami „Odczyt”.

- W tym samym scenariuszu co powyżej, jeśli „X” jednocześnie należy także do grupy hostów 2, do której dostęp jest **zabroniony** dla grupy A lub B, dostęp do „X” będzie **niedostępny**, pomimo dostępu *Odczyt-zapis* do grupy hostów 1.
- Jeśli grupa A nie ma zdefiniowanych żadnych uprawnień, a grupa B ma dostęp *Odczyt-zapis* do grupy hostów 1, użytkownik otrzyma dostęp **Odczyt-zapis** do „X”.
- Jeśli grupa A ma dostęp *Odmowa* do grupy hostów 1, a grupa B ma dostęp *Odczyt-zapis* do grupy hostów 1, dostęp użytkownika do „X” będzie **zabroniony**.

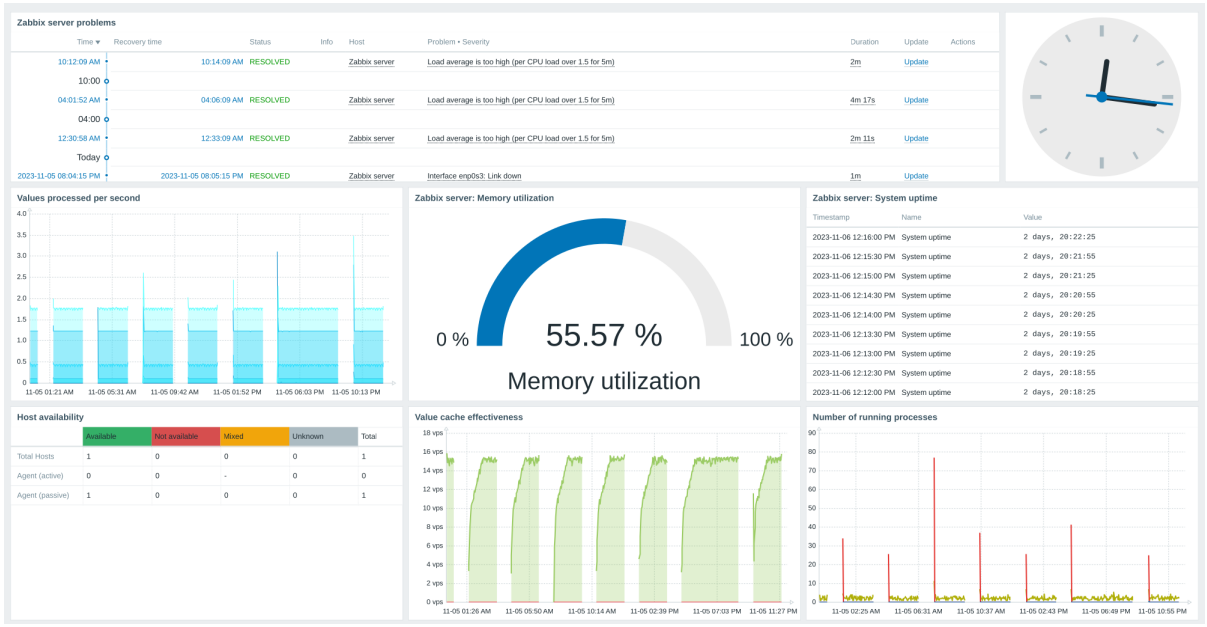
Inne szczegóły

- Użytkownik na poziomie Admin z uprawnieniami *Read-write* do hosta nie będzie mógł podłączać/odłączać szablonów, jeśli nie ma dostępu do grupy szablonów, do której one należą. Przy uprawnieniach *Read* do grupy szablonów będzie mógł podłączać/odłączać szablony do hosta, jednak nie będzie widział żadnych szablonów na liście szablonów i nie będzie mógł wykonywać operacji na szablonach w innych miejscach.
- Użytkownik na poziomie Admin z uprawnieniami *Read* do hosta nie zobaczy hosta na liście hostów w sekcji konfiguracji; jednak wyzwalacze hosta będą dostępne w konfiguracji usług IT.
- Każdy użytkownik niebędący Super Adminem (w tym „guest”) może widzieć mapy sieci, o ile mapa jest pusta lub zawiera tylko obrazy. Gdy do mapy zostaną dodane hosty, grupy hostów lub wyzwalacze, uprawnienia będą respektowane.
- Serwer Zabbix nie będzie wysyłał powiadomień do użytkowników zdefiniowanych jako odbiorcy operacji akcji, jeśli dostęp do danego hosta jest jawnie „denied”.

Zaplanowane raporty

Omówienie

Funkcja *Raporty cykliczne* umożliwia skonfigurowanie wysyłania wersji PDF danego dashboardu do określonych odbiorców w regularnych odstępach czasu.



Wymagania wstępne:

- Usługa sieciowa Zabbix musi być zainstalowana i prawidłowo skonfigurowana, aby umożliwić generowanie raportów cyklicznych — instrukcje znajdziesz w sekcji Konfigurowanie raportów cyklicznych.
- Użytkownik musi mieć rolę użytkownika typu *Administrator* lub *Superadministrator* z następującymi uprawnieniami:
  - *Raporty cykliczne* w bloku *Dostęp do elementów interfejsu* (aby wyświetlać ustawienia raportów)
  - *Zarządzanie raportami cyklicznymi* w bloku *Dostęp do działań* (aby tworzyć i edytować raporty)

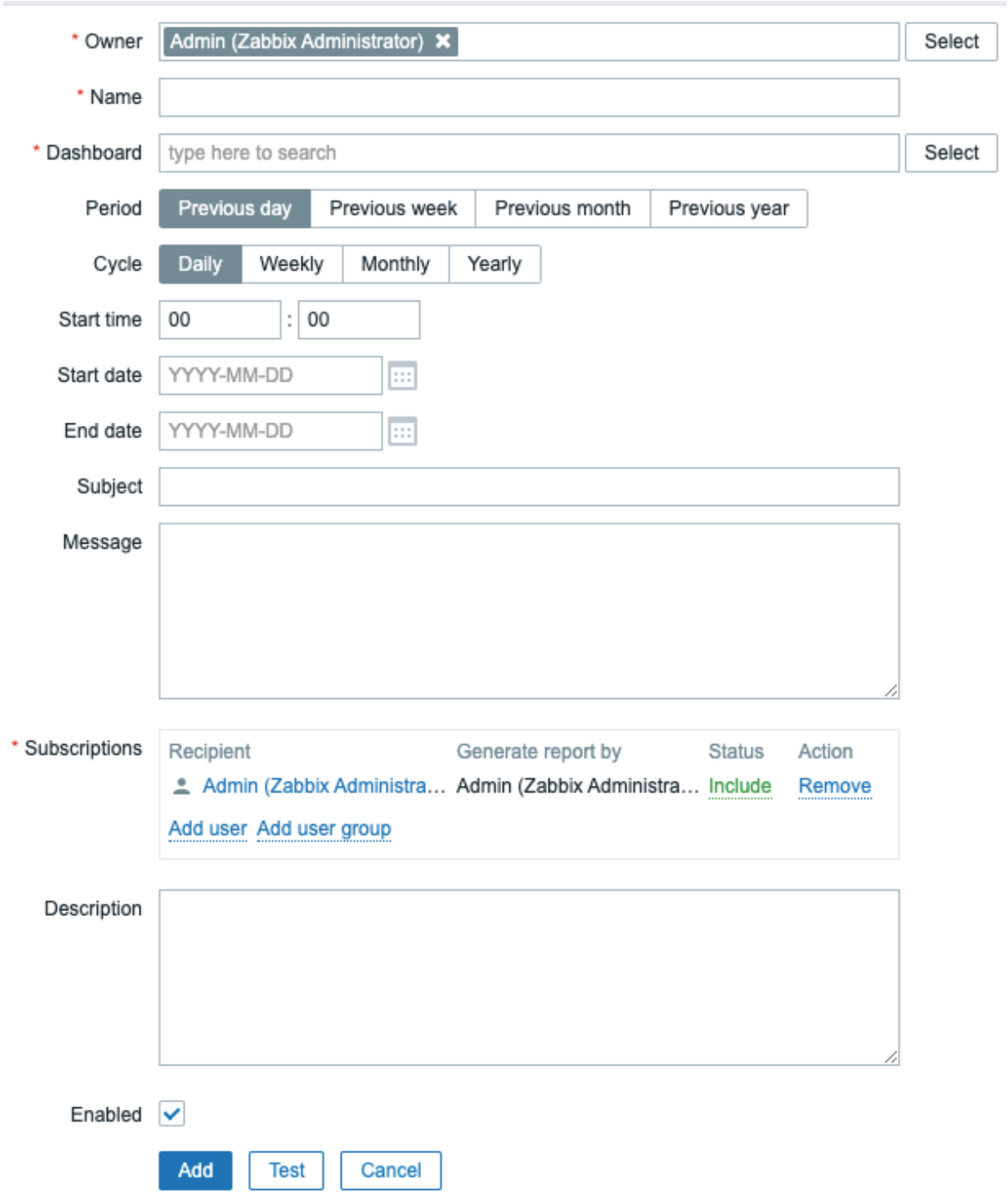
Aby utworzyć raport cykliczny w frontend Zabbix, wykonaj następujące czynności:

1. Przejdź do sekcji *Raporty* > *Raporty cykliczne*.
2. Kliknij przycisk *Utwórz raport* w prawym górnym rogu ekranu.
3. Wprowadź parametry raportu w formularzu.

Możesz również utworzyć raport, otwierając istniejący raport, klikając przycisk *Klonuj* i zapisując go pod inną nazwą.

Konfiguracja

Karta *Raporty zaplanowane* zawiera ogólne atrybuty raportu.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Właściciel	Użytkownik, który tworzy raport. Użytkownicy z poziomem *Super admin* mogą zmienić właściciela. W przypadku użytkowników z poziomem *Admin* to pole jest tylko do odczytu.
Nazwa	Nazwa raportu; musi być unikatowa.
Dashboard	Dashboard, na podstawie którego tworzony jest raport; jednocześnie można wybrać tylko jeden dashboard. Aby wybrać dashboard, zacznij wpisywać jego nazwę - zostanie wyświetlona lista pasujących dashboardów; przewiń listę, aby wybrać dashboard. Możesz również kliknąć *Wybierz* obok pola i wybrać dashboard z wyświetlonej listy.
Okres	Okres, za który zostanie przygotowany raport. Wybierz poprzedni dzień, tydzień, miesiąc lub rok.
Cykl	Częstotliwość generowania raportu. Raporty mogą być wysyłane codziennie, co tydzień, co miesiąc lub co rok. Tryb „Tygodniowo” umożliwia wybranie dni tygodnia, w które raport będzie wysyłany.
Godzina rozpoczęcia	Godzina w formacie hh:mm, o której raport zostanie przygotowany. Należy pamiętać, że zostanie użyta strefa czasowa serwera Zabbix.

Parametr	Opis
Powtarzaj w	Dni tygodnia, w które raport będzie wysyłany. To pole jest dostępne tylko wtedy, gdy dla opcji *Cykl* ustawiono wartość „Tygodniowo”.
Data rozpoczęcia	Data rozpoczęcia regularnego generowania raportu.
Data zakończenia	Data zakończenia regularnego generowania raportu.
Temat	Temat wiadomości e-mail z raportem. Obsługiwane makra: {TIME}, {TIMESTAMP}.
Wiadomość	Treść wiadomości e-mail z raportem. Obsługiwane makra: {TIME}, {TIMESTAMP}.
Subskrypcje	Lista odbiorców raportu. Domyślnie obejmuje tylko właściciela raportu. Odbiorcą raportu może być dowolny użytkownik Zabbix ze skonfigurowanym nośnikiem e-mail. Kliknij *Dodaj użytkownika* lub *Dodaj grupę użytkowników*, aby dodać kolejnych odbiorców. Kliknij nazwę użytkownika, aby edytować ustawienia: *Generuj raport na podstawie* - określa, czy dane raportu powinny być generowane na podstawie uprawnień do dashboardu bieżącego użytkownika czy odbiorcy. *Status* - wybierz „Uwzględnij”, aby wysłać raport użytkownikowi, lub „Wyklucz”, aby uniemożliwić wysłanie raportu temu użytkownikowi. Co najmniej jeden użytkownik musi mieć status „Uwzględnij”. Status „Wyklucz” może służyć do wykluczania określonych użytkowników z uwzględnionej grupy użytkowników. Należy pamiętać, że użytkownicy z niewystarczającymi uprawnieniami (czyli użytkownicy z rolą opartą na typie użytkownika *Admin*, którzy nie są członkami tej samej grupy użytkowników co odbiorca lub właściciel raportu) zobaczą tekst „Niedostępny użytkownik” lub „Niedostępna grupa użytkowników” zamiast rzeczywistych nazw w polach *Odbiorca* i *Generuj raport na podstawie*; pola *Status* i *Akcja* będą wyświetlane jako tylko do odczytu.
Włączony	Status raportu. Wyczyszczenie tego pola wyboru wyłączy raport.
Opis	Opcjonalny opis raportu. Opis jest przeznaczony do użytku wewnętrznego i nie zostanie wysłany odbiorcom raportu.

Przyciski formularza

Przyciski znajdujące się u dołu formularza umożliwiają wykonanie kilku operacji.

Add	Dodaj raport. Ten przycisk jest dostępny tylko dla nowych raportów.
Update	Zaktualizuj właściwości raportu.
Clone	Utwórz kolejny raport na podstawie właściwości bieżącego raportu.
Test	Sprawdź, czy konfiguracja raportu jest poprawna, wysyłając raport do bieżącego użytkownika.
Delete	Usuń raport.
Cancel	Anuluj edycję właściwości raportu.

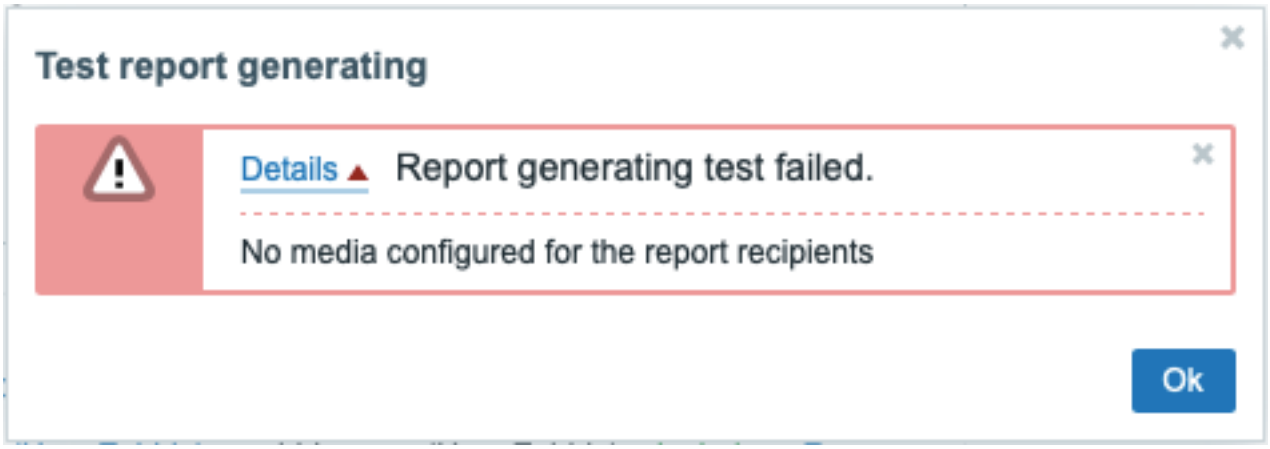
Testowanie

Aby przetestować raport, kliknij przycisk *Test* u dołu formularza konfiguracji raportu.

Note:
Przycisk *Test* nie jest dostępny, jeśli formularz konfiguracji raportu został otwarty z menu działań pulpitu nawigacyjnego.

Jeśli konfiguracja jest poprawna, raport testowy zostanie natychmiast wysłany do bieżącego użytkownika. W przypadku raportów testowych ustawienia subskrybentów i użytkownika *Generate report by* są ignorowane.

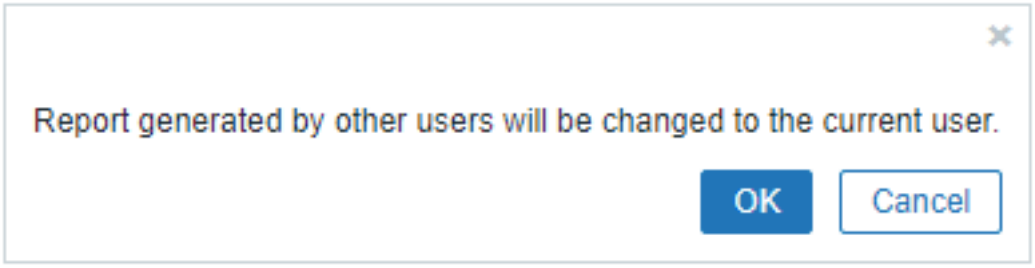
Jeśli konfiguracja jest niepoprawna, zostanie wyświetlony komunikat o błędzie opisujący możliwą przyczynę.



Aktualizacja raportu

Aby zaktualizować istniejący raport, kliknij nazwę raportu, wprowadź wymagane zmiany konfiguracyjne, a następnie kliknij przycisk *Update*.

Jeśli istniejący raport zostanie zaktualizowany przez innego użytkownika i ten użytkownik zmieni Dashboard, po kliknięciu przycisku *Update* zostanie wyświetlony komunikat ostrzegawczy „Raport generowany przez innych użytkowników zostanie zmieniony na bieżącego użytkownika".



Kliknięcie *OK* na tym etapie spowoduje następujące zmiany:

- Ustawienia *Generate report by* zostaną zaktualizowane tak, aby wyświetlać użytkownika, który edytował raport jako ostatni (chyba że *Generate report by* jest ustawione na odbiorcę).
- Użytkownicy wyświetlani jako „Inaccessible user" lub „Inaccessible user group" zostaną usunięci z listy subskrybentów raportu.

Kliknięcie *Cancel* spowoduje zamknięcie formularza konfiguracji i anulowanie aktualizacji raportu.

Klonowanie raportu

Aby szybko sklonować istniejący raport, kliknij przycisk *Clone* u dołu formularza konfiguracji istniejącego raportu. W przypadku klonowania raportu utworzonego przez innego użytkownika bieżący użytkownik staje się właścicielem nowego raportu.

Ustawienia raportu zostaną skopiowane do nowego formularza konfiguracji raportu z uwzględnieniem uprawnień użytkownika:

- Jeśli użytkownik klonujący raport nie ma uprawnień do pulpitu, pole *Dashboard* zostanie wyczyszczone.
- Jeśli użytkownik klonujący raport nie ma uprawnień do niektórych użytkowników lub grup użytkowników na liście *Subscriptions*, niedostępni odbiorcy nie zostaną sklonowani.
- Ustawienia *Generate report by* zostaną zaktualizowane tak, aby wyświetlać bieżącego użytkownika (chyba że w polu *Generate report by* ustawiono odbiorcę).

Zmień wymagane ustawienia i nazwę raportu, a następnie kliknij *Add*.

Eksport danych

Przegląd

Zabbix obsługuje eksport danych w czasie rzeczywistym na dwa sposoby:

- eksport do plików
- przesyłanie strumieniowe do systemów zewnętrznych

Można eksportować następujące encje:

- zdarzenia wyzwalaczy
- wartości pozycji
- trendy (eksport tylko do plików)

Eksport do plików

Przegląd

Możliwe jest skonfigurowanie eksportu w czasie rzeczywistym zdarzeń wyzwalaczy, wartości pozycji i trendów w formacie JSON rozdzielanym znakami nowej linii.

Eksport jest wykonywany do plików, gdzie każda linia pliku eksportu jest obiektem JSON. Mapowania wartości nie są stosowane.

W przypadku błędów (danych nie można zapisać do pliku eksportu albo pliku eksportu nie można zmienić nazwy lub nie można utworzyć nowego po zmianie nazwy), element danych jest odrzucany i nigdy nie jest zapisywany do pliku eksportu. Jest zapisywany wyłącznie w bazie danych Zabbix. Zapisywanie danych do pliku eksportu zostaje wznowione, gdy problem z zapisem zostanie rozwiązany.

Plik eksportu jest tworzony z uprawnieniami odczytu i zapisu tylko dla właściciela pliku. Dodatkowo plik jest czytelny dla grupy właściciela. Wszystkie pozostałe uprawnienia są zabronione.

Należy pamiętać, że host/pozycja może nie mieć metadanych (grup hostów, nazwy hosta, nazwy pozycji), jeśli host/pozycja została usunięta po odebraniu danych, ale przed wyeksportowaniem danych przez serwer.

Konfiguracja

Eksport w czasie rzeczywistym zdarzeń wyzwalaczy, wartości pozycji i trendów konfiguruje się przez określenie katalogu dla plików eksportu — zobacz parametr ExportDir w konfiguracji serwera.

Dostępne są dwa inne parametry:

- Parametr `ExportFileSize` umożliwia ustawienie maksymalnego dozwolonego rozmiaru pojedynczego pliku eksportu. Gdy proces musi zapisać dane do pliku, najpierw sprawdza jego rozmiar. Jeśli przekracza on skonfigurowany limit, plik otrzymuje nazwę z rozszerzeniem `.old`, a następnie tworzony jest nowy plik z pierwotną nazwą.

Attention:
Dla każdego procesu zapisującego dane zostanie utworzony osobny plik (tj. około 4–30 plików). Ponieważ domyślny rozmiar pliku eksportu wynosi 1G, przechowywanie dużych plików eksportu może szybko wyczerpać miejsce na dysku.

- Parametr `ExportType` umożliwia określenie, które typy encji (zdarzenia, historia, trendy) będą eksportowane.

Protokół eksportu JSON z rozdzieleniem nowymi liniami

Ta sekcja przedstawia szczegóły protokołu eksportu w formacie JSON z rozdzieleniem nowymi liniami, używanego w:

- eksport danych do plików
- strumieniowaniu do systemów zewnętrznych

Wszystkie pliki mają rozszerzenie `.ndjson`. Każda linia pliku eksportu jest obiektem JSON.

Eksport zdarzenia wyzwalacza

Następujące informacje są eksportowane dla zdarzenia problemu:

Field	Type	Description
clock	number	Liczba sekund od Epoki do momentu wykrycia problemu (część całkowita).
ns	number	Liczba nanosekund, które należy dodać do `clock`, aby uzyskać dokładny czas wykrycia problemu.
value	number	1 (zawsze).
eventid	number	ID zdarzenia problemu.
name	string	Nazwa zdarzenia problemu.
severity	number	Ważność zdarzenia problemu (0 - Nie sklasyfikowano, 1 - Informacja, 2 - Ostrzeżenie, 3 - Średni, 4 - Wysoki, 5 - Katastrofa).
hosts	array	Lista hostów uwzględnionych w wyrażeniu wyzwalacza; w tablicy powinien znajdować się co najmniej jeden element.
-	object	
host	string	Nazwa hosta.
name	string	Widoczna nazwa hosta.

Field	Type	Description
groups	array	Lista grup hostów wszystkich hostów uwzględnionych w wyrażeniu wyzwalacza; w tablicy powinien znajdować się co najmniej jeden element.
-	string	Nazwa grupy hostów.
tags	array	Lista tagów problemu (może być pusta).
-	object	
tag	string	Nazwa tagu.
value	string	Wartość tagu (może być pusta).

Następujące informacje są eksportowane dla zdarzenia odzyskania:

Field	Type	Description
clock	number	Liczba sekund od Epoki do momentu rozwiązania problemu (część całkowita).
ns	number	Liczba nanosekund, które należy dodać do `clock`, aby uzyskać dokładny czas rozwiązania problemu.
value	number	0 (zawsze).
eventid	number	ID zdarzenia odzyskania.
p_eventid	number	ID zdarzenia problemu.

Przykłady eksportu zdarzeń wyzwalacza

Problem:

```
{"clock":1519304285,"ns":123456789,"value":1,"name":"Either Zabbix agent is unreachable on Host B or polle
```

Odzyskanie:

```
{"clock":1519304345,"ns":987654321,"value":0,"eventid":43,"p_eventid":42}
```

Problem (generowanie wielu zdarzeń problemu):

```
{"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
```

Odzyskanie:

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":43}
```

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":42}
```

Eksport wartości pozycji

Poniższe informacje są eksportowane dla zebranej wartości pozycji:

Field	Type	Description
host	object	Nazwa hosta hosta pozycji.
host	string	Nazwa hosta.
name	string	Widoczna nazwa hosta.
groups	array	Lista grup hosta hosta pozycji; w tablicy powinien znajdować się co najmniej jeden element.
-	string	Nazwa grupy hostów.
item_tags	array	Lista tagów pozycji (może być pusta).
-	object	
tag	string	Nazwa tagu.
value	string	Wartość tagu (może być pusta).
itemid	number	ID pozycji.
name	string	Widoczna nazwa pozycji.
clock	number	Liczba sekund od początku ery Unix do momentu, w którym wartość została zebrana (część całkowita).
ns	number	Liczba nanosekund, które należy dodać do `clock`, aby uzyskać dokładny czas zebrania wartości.

Field	Type	Description
timestamp (*Log* only)	number	0, jeśli niedostępne.
source (*Log* only)	string	Pusty ciąg, jeśli niedostępne.
severity (*Log* only)	number	0, jeśli niedostępne.
eventid (*Log* only)	number	0, jeśli niedostępne.
value	number (for numeric items) or string (for text items)	Zebrana wartość pozycji.
type	number	Typ zebranej wartości: 0 - liczba zmiennoprzecinkowa, 1 - znak, 2 - log, 3 - liczba całkowita bez znaku, 4 - tekst, 5 - binarne, 6 - JSON

Przykład eksportu wartości pozycji

Wartość numeryczna (bez znaku):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość numeryczna (zmiennoprzecinkowa):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość znakowa, tekstowa:

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość logu:

```
{"host":{"host":"Host A","name":"Host A visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Eksport trendów

Poniższe informacje są eksportowane dla obliczonej wartości trendu:

Field		Type	Description
host		object	Nazwa hosta hosta pozycji.
	host	string	Nazwa hosta.
	name	string	Widoczna nazwa hosta.
groups		array	Lista grup hosta hosta pozycji; w tablicy powinien znajdować się co najmniej jeden element.
	-	string	Nazwa grupy hosta.
item_tags		array	Lista tagów pozycji (może być pusta).
	-	object	
	tag	string	Nazwa tagu.
	value	string	Wartość tagu (może być pusta).
itemid		number	ID pozycji.
name		string	Widoczna nazwa pozycji.
clock		number	Liczba sekund od Ery do momentu, w którym wartość została zebrana (część całkowita).
count		number	Liczba wartości zebranych dla danej godziny.
min		number	Minimalna wartość pozycji dla danej godziny.
avg		number	Średnia wartość pozycji dla danej godziny.
max		number	Maksymalna wartość pozycji dla danej godziny.
type		number	Typ wartości: 0 - liczba zmiennoprzecinkowa, 3 - liczba całkowita bez znaku

Przykłady eksportu trendów

Wartość liczbowa (bez znaku):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Wartość liczbowa (zmiennoprzecinkowa):

```
{"host":{"host":"Host B","name":"Host B visible"},"groups":["Group X","Group Y","Group Z"],"item_tags":[{"
```

Strumieniowanie do systemów zewnętrznych

Przegląd

Możliwe jest przesyłanie wartości pozycji i zdarzeń z Zabbix do systemów zewnętrznych za pośrednictwem HTTP (zobacz szczegóły protokołu).

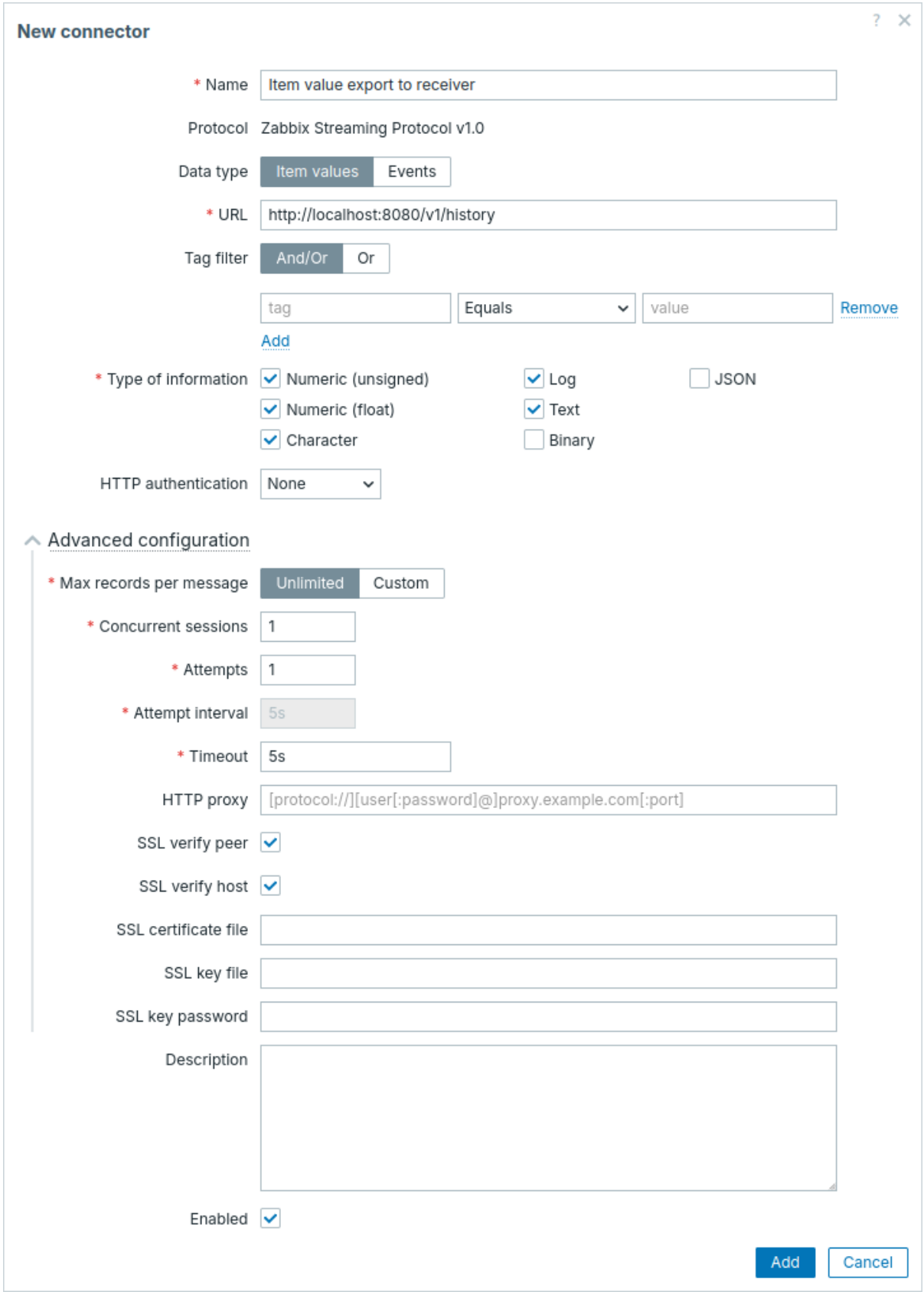
Filtr tagów może służyć do przesyłania podzbiorów wartości pozycji lub zdarzeń.

Za przesyłanie danych odpowiadają dwa typy procesów serwera Zabbix: `connector manager` i `connector worker`. Wewnętrzna pozycja Zabbix `zabbix[connector_queue]` umożliwia monitorowanie liczby wartości umieszczonych w kolejce connectora.

Konfiguracja

Aby skonfigurować strumieniowanie danych do systemu zewnętrznego, należy wykonać następujące czynności:

1. Skonfiguruj system zdalny do odbierania danych z Zabbix. W tym celu dostępne są następujące narzędzia:
   - Przykład prostego odbiornika, który zapisuje odebrane informacje w plikach `events.ndjson` i `history.ndjson`.
   - Kafka connector for Zabbix server - lekki serwer napisany w języku Go, przeznaczony do przekazywania wartości pozycji i zdarzeń z serwera Zabbix do brokera Kafka.
2. Ustaw wymaganą liczbę procesów roboczych connectora w Zabbix, dostosowując parametr `StartConnectors` w pliku `zabbix_server.conf`. Liczba procesów roboczych connectora powinna odpowiadać skonfigurowanej liczbie connectorów w frontend Zabbix (lub ją przekraczać, jeśli liczba jednoczesnych sesji jest większa niż 1). Następnie uruchom ponownie serwer Zabbix.
3. Skonfiguruj nowy connector w frontend Zabbix (*Administration* > *General* > *Connectors*) i przeładuj pamięć podręczną serwera za pomocą polecenia `zabbix_server -R config_cache_reload`.



Pola obowiązkowe są oznaczone gwiazdką.

Parameter	Description
Name	Wprowadź nazwę connectora.
Data type	Wybierz typ danych do strumieniowania: **Item values** - strumieniowanie wartości pozycji z Zabbix do systemów zewnętrznych; **Events** - strumieniowanie zdarzeń z Zabbix do systemów zewnętrznych.
URL	Wprowadź URL odbiornika. Obsługiwane są makra użytkownika.

Parameter	Description
Tag filter	Eksportuj tylko wartości pozycji lub zdarzenia pasujące do filtra tagów. Jeśli filtr nie zostanie ustawiony, eksportowane będą wszystkie dane. Można uwzględniać lub wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnia określone nazwy tagów; **Equals** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Contains** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter); **Does not exist** - wyklucza określone nazwy tagów; **Does not equal** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Does not contain** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Or; **Or** - wystarczy spełnienie jednego warunku.
Type of information	Wybierz typ informacji (liczbowe (bez znaku), liczbowe (zmiennoprzecinkowe), znakowe itd.), według którego mają być filtrowane wartości pozycji strumieniowane przez connector. To pole jest dostępne, jeśli *Data type* ma wartość „Item values”.
HTTP authentication	Wybierz opcję uwierzytelniania: **None** - uwierzytelnianie nie jest używane; **Basic** - używane jest uwierzytelnianie podstawowe; **NTLM** - używane jest uwierzytelnianie NTLM (Windows NT LAN Manager); **Kerberos** - używane jest uwierzytelnianie Kerberos (zobacz także: Konfigurowanie Kerberos z Zabbix); **Digest** - używane jest uwierzytelnianie Digest; **Bearer** - używane jest uwierzytelnianie Bearer.
Username	Wprowadź nazwę użytkownika (do 255 znaków). Obsługiwane są makra użytkownika. To pole jest dostępne, jeśli *HTTP authentication* ma wartość „Basic”, „NTLM”, „Kerberos” lub „Digest”.
Password	Wprowadź hasło użytkownika (do 255 znaków). Obsługiwane są makra użytkownika. To pole jest dostępne, jeśli *HTTP authentication* ma wartość „Basic”, „NTLM”, „Kerberos” lub „Digest”.
Bearer token	Wprowadź token Bearer. Obsługiwane są makra użytkownika. To pole jest dostępne i wymagane, jeśli *HTTP authentication* ma wartość „Bearer”.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić zaawansowane opcje konfiguracji (zobacz poniżej).
Max records per message	Określ maksymalną liczbę wartości lub zdarzeń, które mogą być strumieniowane w ramach jednej wiadomości.
Concurrent sessions	Wybierz liczbę procesów nadawczych uruchamianych dla tego connectora. Można określić do 100 sesji; wartość domyślna to „1”.
Attempts	Liczba prób strumieniowania danych. Można określić do 5 prób; wartość domyślna to „1”.
Attempt interval	Określ, jak długo connector powinien czekać po nieudanej próbie strumieniowania danych. Można określić maksymalnie 10 s; wartość domyślna to „5s”. To pole jest dostępne, jeśli *Attempts* ma wartość „2” lub większą. Nieudane próby to takie, podczas których nie udało się nawiązać połączenia lub kod odpowiedzi HTTP nie wynosi 200, 201, 202, 203 ani 204. Ponowienia są **wyzwalane** w przypadku błędów komunikacji lub gdy kod odpowiedzi HTTP nie wynosi 200, 201, 202, 203, 204, 400, 401, 403, 404, 405, 415 ani 422. Przekierowania są obsługiwane, dlatego 302 -> 200 jest odpowiedzią pozytywną, natomiast 302 -> 503 spowoduje ponowienie.
Timeout	Określ limit czasu wiadomości (1-60 sekund, domyślnie - 5 sekund). Obsługiwane są przyrostki czasu (np. `30s`, `1m`). Obsługiwane są makra użytkownika.

Parameter	Description
HTTP proxy	Można określić proxy HTTP w następującym formacie: `[protocol://][username[:password]@]proxy.example.com[:port]` Obsługiwane są makra użytkownika. Opcjonalny prefiks `protocol://` może służyć do określenia alternatywnych protokołów proxy (obsługa prefiksu protokołu została dodana w cURL 7.21.7). Jeśli protokół nie zostanie określony, proxy będzie traktowane jako proxy HTTP. Domyślnie używany jest port 1080. Jeśli określono *HTTP proxy*, proxy zastąpi zmienne środowiskowe związane z proxy, takie jak `http_proxy` i `HTTPS_PROXY`. Jeśli nie zostanie określone, proxy nie zastąpi zmiennych środowiskowych związanych z proxy. Wprowadzona wartość jest przekazywana bez zmian, bez sprawdzania poprawności. Można również wprowadzić adres proxy SOCKS. Jeśli określony zostanie nieprawidłowy protokół, connector nie będzie mógł strumieniować wartości pozycji ani zdarzeń z Zabbix. Należy pamiętać, że w przypadku proxy HTTP obsługiwane jest tylko proste uwierzytelnianie.
SSL verify peer	Zaznacz pole wyboru, aby zweryfikować certyfikat SSL serwera WWW. Certyfikat serwera jest weryfikowany względem certyfikatów zaufanych urzędów certyfikacji (CA) znajdujących się w systemowej lokalizacji CA. Lokalizację plików CA można zmienić za pomocą parametru konfiguracji serwera Zabbix lub proxy `SSLCALocation`.
SSL verify host	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub *Subject Alternate Name* certyfikatu serwera WWW jest zgodne. Ustawia to opcję cURL `CURLOPT_SSL_VERIFYHOST`.
SSL certificate file	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM[1]. Obsługiwane są makra użytkownika. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole *SSL key file* puste. Jeśli klucz jest zaszyfrowany, określ hasło w polu *SSL key password*. Katalog zawierający ten plik jest określany przez parametr konfiguracji serwera Zabbix lub proxy `SSLCertLocation`.
SSL key file	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta. Plik klucza prywatnego musi być w formacie PEM[1]. Obsługiwane są makra użytkownika. Katalog zawierający ten plik jest określany przez parametr konfiguracji serwera Zabbix lub proxy `SSLKeyLocation`.
SSL key password	Hasło pliku klucza prywatnego SSL. Obsługiwane są makra użytkownika.
Description	Wprowadź opis connectora.
Enabled	Zaznacz pole wyboru, aby włączyć connector.

Jeśli connector Kafka zostanie skonfigurowany z rozdzielaną przecinkami listą adresów brokerów rozruchowych (na przykład `Kafka.URL=kafka1.example.com:9093,kafka2.example.com:9093`), klient Kafka połączy się z brokerem lub brokerami, które odpowiedzą jako pierwsze, i użyje ich metadanych klastra. Jeśli lista zawiera adresy z różnych klastrów Kafka, używany będzie tylko klaster odpowiadający najszybciej, a pozostałe adresy zostaną zapisane w dzienniku jako niedostępne; w rezultacie mogą pojawić się ostrzeżenia podczas uruchamiania, takie jak poniższe, nawet jeśli connector jest połączony:

```
kafka cluster connected, but broker(s) "kafka1.example.com:9093, kafka2.example.com:9093" unavailable; wil
```

W niektórych środowiskach (sieciach prywatnych, sieciach kontenerowych lub konfiguracjach niestandardowego DNS/hosts) nazwy hostów lub adresy IP mogą być rozwiązywane do adresów pętli zwrotnej (na przykład `127.0.0.1/localhost`) albo normalizowane przez klienta, co może powodować, że takie ostrzeżenia będą mylące. Aby ograniczyć niejasności, upewnij się, że wszystkie adresy `Kafka.URL` należą do tego samego klastra Kafka, sprawdź rozwiązywanie DNS z hosta connectora oraz wartości `advertised.listeners` brokerów i preferuj adresy rozwiązywane do adresu rozgłaszanego przez brokera.

Protokół

Komunikacja między serwerem a odbiornikiem odbywa się za pośrednictwem HTTP z użyciem REST API, NDJSON oraz "Content-Type: application/x-ndjson".

Więcej informacji można znaleźć w sekcji Newline-delimited JSON export protocol.

Żądanie serwera

Przykład strumieniowania wartości pozycji:

```
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
```

Przykład strumieniowania zdarzeń:

```
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}
```

Odpowiedź odbiorcy

Odpowiedź składa się z kodu statusu odpowiedzi HTTP oraz ładunku JSON. Kod statusu odpowiedzi HTTP musi mieć wartość ”200”, ”201”, ”202”, ”203” lub ”204” dla żądań, które zostały obsłużone pomyślnie, inny dla żądań zakończonych niepowodzeniem.

Przykład powodzenia:

```
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"}
```

Przykład z błędami:

```
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"}
```

Brama SNMP

Przegląd

Zabbix SNMP gateway to rozszerzenie AgentX dla snmpd, obsługujące zarówno odpytywanie SNMP, jak i pułapki SNMP.

Za pomocą Zabbix SNMP gateway można używać protokołu SNMP do pobierania:

- danych wyzwalacza;
- danych wyzwalacza problemu;
- statusu grupy hostów (liczby wyzwalaczy według stanu wyzwalacza w każdej grupie)

Dane są pobierane za pomocą OID, który jest kombinacją wspólnej bazy i określonego sufiksu. Wspólna **baza** jest ustawiana w pliku konfiguracyjnym SNMP gateway, na przykład:

- `BaseOID=1.3.6.1.4.1.3043.7.55` - dla dowolnych danych wyzwalacza;
- `ProblemBaseOID=1.3.6.1.4.1.3047.7.55` - dla danych wyzwalacza problemu;
- `BaseOID=1.3.6.1.4.1.3046.7.55` - dla statusu grupy hostów.

Sufiks OID jest ustawiany w konfiguracji wyzwalaczy hosta jako **tag** (na przykład `OIDSuffix:3`) we frontend.

W takim przypadku wszystkie informacje o wyzwalaczu będą dostępne pod `OID=1.3.6.1.4.1.3043.7.55.X.3`. „X” oznacza tutaj numer pola danych wyzwalacza (tj. 1 - sufiks, 2 - ID, 3 - wyrażenie, 4 - opis itd.).

Bardziej szczegółowy opis oraz przykład pliku konfiguracyjnego można znaleźć w pliku readme SNMP gateway.

Instalacja i konfiguracja

Instrukcje dotyczące:

- instalacji i konfiguracji snmpd;
- włączania obsługi AgentX;
- konfiguracji bramy SNMP Zabbix;
- konfiguracji pułapek SNMP dla zmian stanu wyzwalacza

znajdują się w pliku readme bramy SNMP.

Pobieranie danych

Po poprawnym skonfigurowaniu wszystkiego możesz użyć poleceń `snmpwalk` i `snmpget`, aby pobrać dane:

```
[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
...
```

Opcje filtrowania

Możesz ograniczyć informacje o problemach wyzwalaczy w konfiguracji bramy SNMP:

- według ważności (domyślnie `ProblemMinSeverity=-1`)
- przez ukrywanie potwierdzonych problemów (domyślnie `ProblemHideAck=false`)

Możesz ograniczyć liczbę problemów na grupę hostów w konfiguracji bramy SNMP:

- według wyzwalaczy w stanie nieznanym (domyślnie `CountUnknown=false`)
- według wyzwalaczy z potwierdzonymi/niepotwierdzonymi/wszystkimi problemami (domyślnie `CountAcknowledgeStatus=all`)

Eksport/import konfiguracji

Omówienie

Funkcja eksportu/importu Zabbixa umożliwia wymianę różnych obiektów konfiguracji między jednym systemem Zabbix a innym.

Typowe przypadki użycia tej funkcji:

- udostępnianie szablonów lub map sieciowych - użytkownicy Zabbixa mogą udostępniać swoje parametry konfiguracji
- przesłanie szablonu do Zabbix Community templates. Następnie inni mogą pobrać szablon i zaimportować plik do Zabbixa.
- integracja z narzędziami firm trzecich - uniwersalne formaty YAML, XML i JSON umożliwiają integrację oraz import/eksport danych z narzędziami i aplikacjami firm trzecich

Co można eksportować/importować

Obiekty, które można eksportować/importować, to:

- Grupy hostów (*tylko za pośrednictwem Zabbix API*)
- Grupy szablonów (*tylko za pośrednictwem Zabbix API*)
- Szablony
- Hosty
- Mapy sieci
- Typy mediów

- Obrazy
- Pulpity
- Wyrażenia regularne

Format eksportu

Dane można eksportować za pomocą frontend Zabbix lub Zabbix API. Obsługiwane formaty eksportu to YAML, XML i JSON.

Szczegóły dotyczące eksportu

- Wszystkie obsługiwane elementy są eksportowane do jednego pliku.
- Encje hosta i szablonu (pozycje, wyzwalacze, wykresy, reguły wykrywania), które są dziedziczone z powiązanych szablonów, nie są eksportowane. Wszelkie zmiany wprowadzone w tych encjach na poziomie hosta (takie jak zmieniony interwał pozycji, zmodyfikowane wyrażenie regularne lub dodane prototypy do reguły wykrywania niskiego poziomu) zostaną utracone podczas eksportu; podczas importu wszystkie encje z powiązanych szablonów są odtwarzane tak jak w oryginalnym powiązanym szablonie.
- Encje utworzone przez wykrywanie niskiego poziomu oraz wszelkie encje od nich zależne nie są eksportowane. Na przykład wyzwalacz utworzony dla pozycji wygenerowanej przez regułę LLD nie zostanie wyeksportowany.
- Gdy eksportowany host/szablon zawiera encje obsługujące limity czasu, wartości limitów czasu zostaną wyeksportowane, jeśli dla tych encji skonfigurowano własne limity czasu.

Szczegóły dotyczące importu

- Import zatrzymuje się przy pierwszym błędzie.
- Podczas aktualizacji istniejących obrazów w trakcie importu obrazów pole ”imagetype” jest ignorowane, tzn. nie można zmienić typu obrazu za pomocą importu.
- Podczas importowania hostów/szablonów z użyciem opcji ”Delete missing” makra hosta/szablonu, których nie ma w pliku importu, zostaną usunięte z hosta/szablonu po imporcie.
- Puste tagi dla pozycji, wyzwalaczy, wykresów, discoveryRules, itemPrototypes, triggerPrototypes, graphPrototypes nie mają znaczenia, tzn. jest tak, jakby ich nie było.
- Jeśli encje importowanego hosta/szablonu mają skonfigurowane własne limity czasu, zostaną one zastosowane; w przeciwnym razie zostaną zastosowane limity czasu proxy/globalne.
- Import obsługuje YAML, XML i JSON, a plik importu musi mieć poprawne rozszerzenie: .yaml i .yml dla YAML, .xml dla XML oraz .json dla JSON. Informacje o obsługiwanych wersjach XML znajdują się w informacjach o zgodności.
- Import obsługuje pliki konfiguracyjne wyłącznie w kodowaniu UTF-8 (z BOM lub bez niego); inne kodowania (UTF16LE, UTF16BE, UTF32LE, UTF32BE itd.) spowodują błąd konwersji importu.

Podstawowy format YAML

Format eksportu YAML zawiera następujące węzły:

- Węzeł główny dla eksportu YAML Zabbix
- Wersja eksportu

```
zabbix_export:
  version: '8.0'
```

Pozostałe węzły zależą od eksportowanych obiektów.

Format XML

Format eksportu XML zawiera następujące tagi:

- Domyślny nagłówek dla dokumentów XML
- Tag główny dla eksportu XML Zabbix
- Wersja eksportu

```
<?xml version="1.0" encoding="UTF-8"?>
<zabbix_export>
    <version>8.0</version>
</zabbix_export>
```

Pozostałe tagi zależą od eksportowanych obiektów.

Format JSON

Format eksportu JSON zawiera następujące obiekty:

- Obiekt główny dla eksportu JSON Zabbix
- Wersja eksportu

```
{
    "zabbix_export": {
        "version": "8.0"
    }
}
```

Pozostałe obiekty zależą od eksportowanych obiektów.

Grupy szablonów

Omówienie

W frontendzie grupy szablonów można eksportować tylko wraz z eksportem szablonu. Gdy szablon jest eksportowany, wszystkie grupy, do których należy, są eksportowane automatycznie.

API umożliwia eksportowanie grup szablonów niezależnie od szablonów.

Format eksportu

```
template_groups:
  - uuid: 36bff6c29af64692839d077febfc7079
    name: 'Network devices'
```

Eksportowane elementy

Element	Type	Description
uuid	string	Unikalny identyfikator tej grupy szablonów.
name	string	Nazwa grupy.

Grupy hostów

Omówienie

W frontendzie grupy hostów można eksportować tylko razem z eksportem hosta. Gdy host jest eksportowany, wszystkie grupy, do których należy, są eksportowane automatycznie razem z nim.

API umożliwia eksportowanie grup hostów niezależnie od hostów.

Format eksportu

```
host_groups:
  - uuid: 6f6799aa69e844b4b3918f779f2abf08
    name: 'Serwery Zabbix'
```

Eksportowane elementy

Element	Type	Description
uuid	string	Unikalny identyfikator tej grupy hostów.
name	string	Nazwa grupy.

Szablony

Omówienie

Szablony są eksportowane wraz z wieloma powiązanymi obiektami i relacjami między obiektami.

Eksport szablonu zawiera:

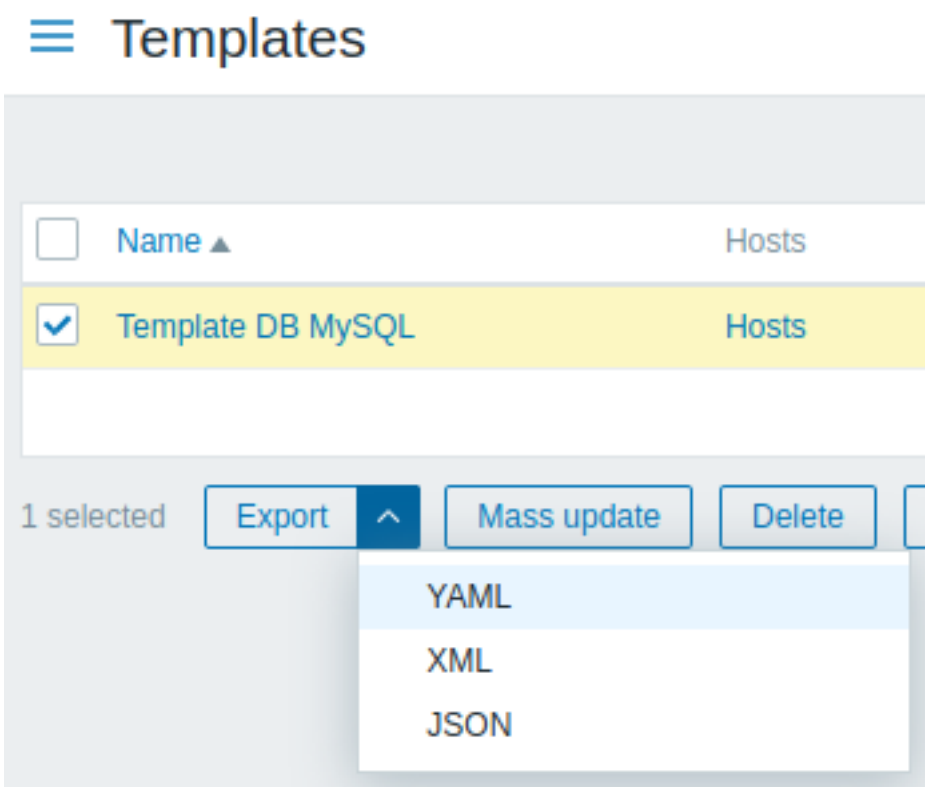
- Powiązane grupy szablonów
- Powiązane grupy hostów (jeśli są używane w konfiguracji prototypu hosta)
- Dane szablonu
- Powiązania z innymi szablonami
- Powiązania z grupami szablonów
- Bezpośrednio powiązane pozycje
- Bezpośrednio powiązane wyzwalacze

- Bezpośrednio powiązane wykresy
- Bezpośrednio powiązane pulpity
- Bezpośrednio powiązane reguły wykrywania ze wszystkimi prototypami
- Bezpośrednio powiązane scenariusze web
- Mapy wartości

Eksportowanie

Aby wyeksportować szablony, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Templates*.
2. Zaznacz pola wyboru szablonów do wyeksportowania.
3. Kliknij *Export* pod listą, aby wyeksportować w domyślnym formacie YAML, lub kliknij strzałkę, aby wybrać inny format eksportu.



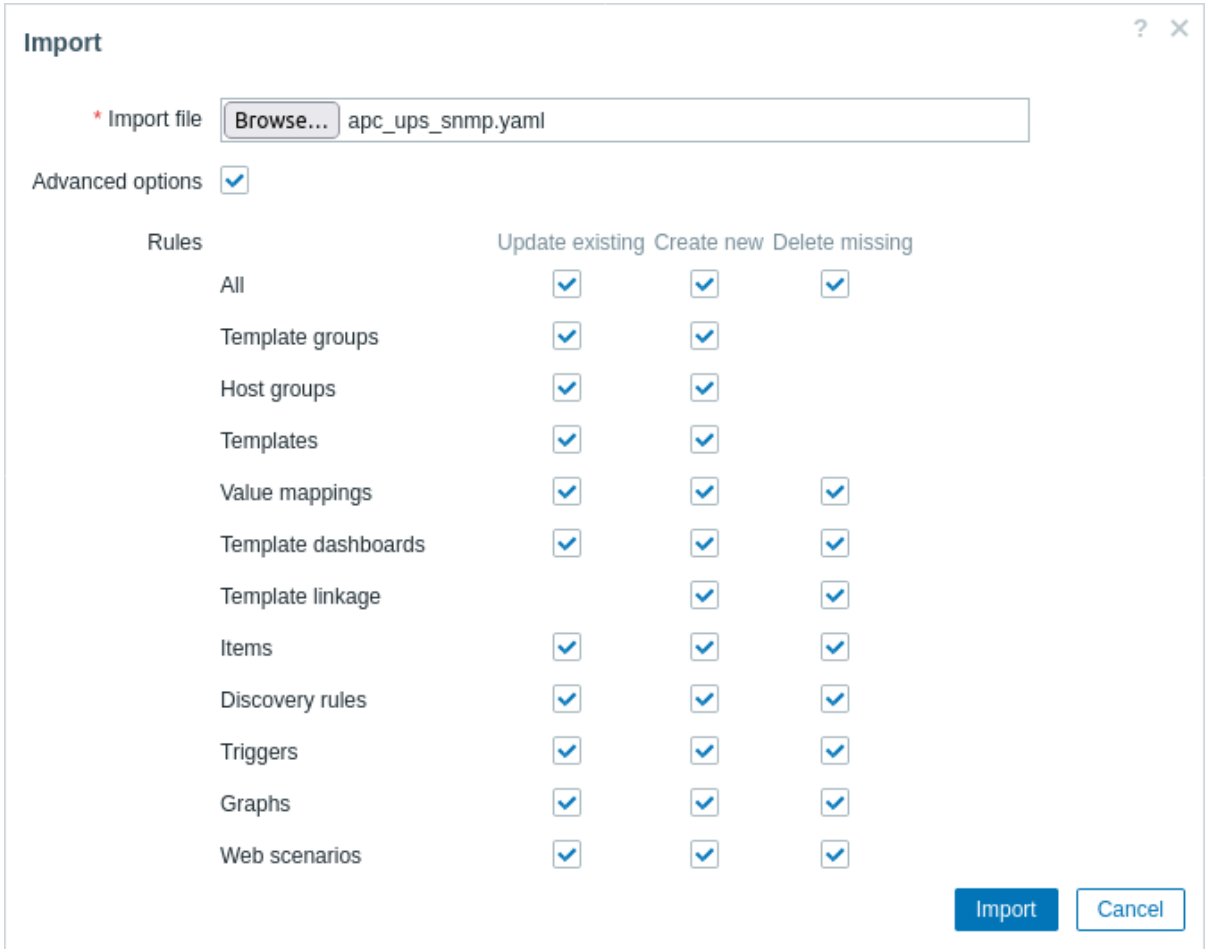
W zależności od wybranego formatu szablony są eksportowane do lokalnego pliku o domyślnej nazwie:

- `zabbix_export_templates.yaml` - w eksporcie YAML;
- `zabbix_export_templates.xml` - w eksporcie XML;
- `zabbix_export_templates.json` - w eksporcie JSON.

Importowanie

Aby zaimportować szablony, wykonaj następujące czynności:

1. Przejdź do sekcji *Zbieranie danych* > *Szablony*.
2. Kliknij przycisk *Importuj* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz pola wyboru reguł importu, które chcesz zastosować (zobacz poniżej).
5. Kliknij przycisk *Importuj* w prawym dolnym rogu formularza konfiguracji.



Jeśli zaznaczysz pole wyboru *Opcje zaawansowane*, zostanie wyświetlona szczegółowa lista wszystkich elementów, które można zaimportować - zaznacz lub odznacz poszczególne reguły importu zgodnie z wymaganiami.

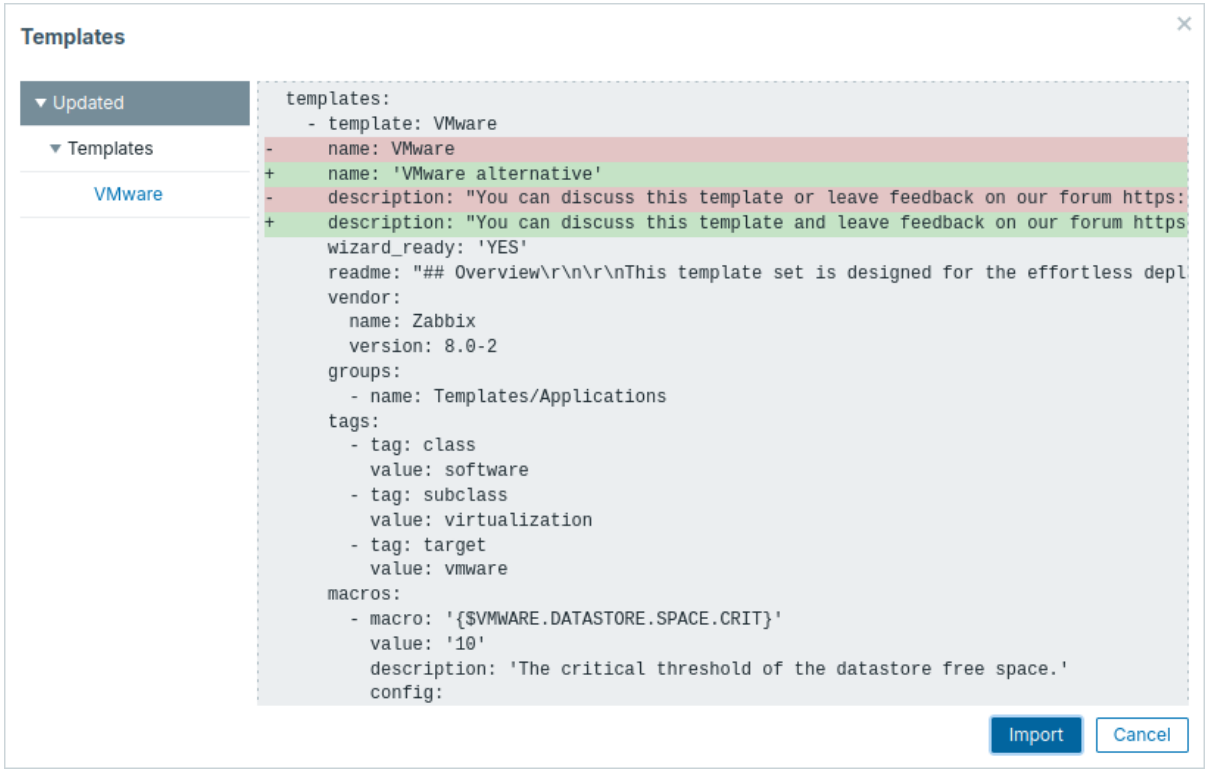
Jeśli klikniesz pole wyboru w wierszu *Wszystkie*, wszystkie znajdujące się poniżej elementy zostaną zaznaczone lub odznaczone.

Reguły importu:

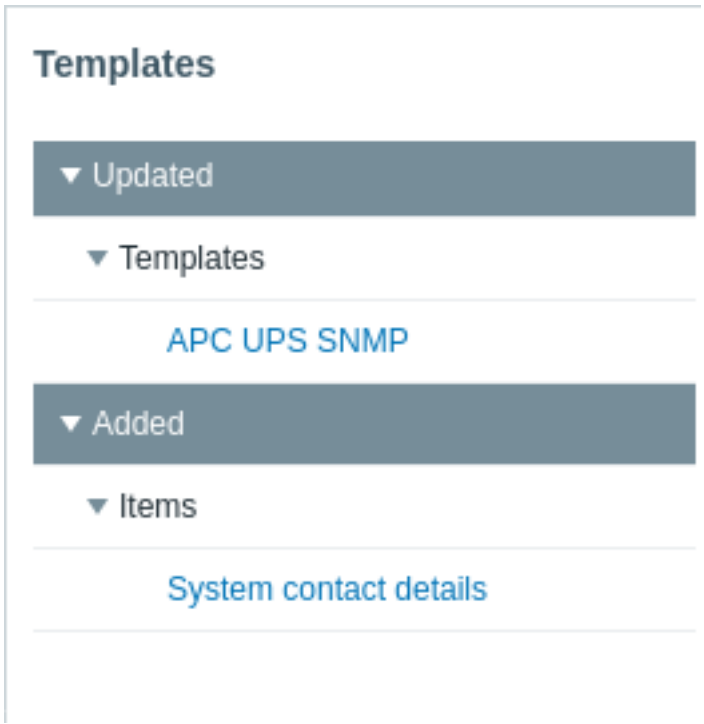
Reguła	Opis
Aktualizuj istniejące	Istniejące elementy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Utwórz nowe	Nowe elementy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.
Usuń brakujące	Istniejące elementy, których nie ma w pliku importu, zostaną usunięte. W przeciwnym razie nie zostaną usunięte. Jeśli opcja *Usuń brakujące* zostanie zaznaczona dla *Powiązania szablonów*, bieżące powiązania szablonów, których nie ma w pliku importu, zostaną usunięte. Encje (pozycje, wyzwalacze, wykresy itd.) odziedziczone z odłączonych szablonów nie zostaną usunięte (chyba że opcja *Usuń brakujące* zostanie również wybrana dla każdej encji).

Attention:
Jeśli szablony o tych samych nazwach już istnieją, podczas importowania należy zaznaczyć opcje *Usuń brakujące*, aby uzyskać czysty import. W ten sposób stare pozycje, których nie ma już w zaktualizowanym szablonie, zostaną usunięte (należy pamiętać, że oznacza to utratę historii tych starych pozycji).

Na następnym ekranie będzie można wyświetlić zawartość importowanego szablonu. Jeśli jest to nowy szablon, wszystkie elementy zostaną wyświetlone na zielono. Jeśli aktualizujesz istniejący szablon, nowe elementy szablonu zostaną wyróżnione na zielono, usunięte elementy szablonu zostaną wyróżnione na czerwono, a elementy, które nie uległy zmianie, zostaną wyświetlone na szarym tle.



Menu po lewej stronie służy do poruszania się po liście zmian. Sekcja *Zaktualizowane* wyróżnia wszystkie zmiany wprowadzone w istniejących elementach szablonu. Sekcja *Dodane* zawiera listę nowych elementów szablonu. Elementy w każdej sekcji są pogrupowane według typu elementu; kliknij szarą strzałkę, aby rozwinąć lub zwinąć grupę elementów.



Przejrzyj zmiany w szablonie, a następnie kliknij przycisk *Importuj*, aby wykonać import szablonu. Komunikat o pomyślnym lub nieudanym imporcie zostanie wyświetlony w frontend.

Format eksportu Gdy szablon jest eksportowany, Zabbix generuje uporządkowany format YAML, JSON lub XML. Eksport obejmuje elementy szablonu, takie jak metadane szablonu, pozycje, makra, wyzwalacze, pulpity i inne.

Każdy element pełni określoną funkcję i może zawierać elementy zagnieżdżone.

W kolejnych sekcjach opisano każdy element w formacie eksportu. W przykładach użyto szablonu Linux by Zabbix agent.

Wielokropek (`...`) oznacza elementy pominięte ze względu na zwięzłość. Uwaga (`see table below`) jest używana, gdy element jest wyjaśniony bardziej szczegółowo w późniejszych sekcjach.

```
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)
```

Metadane szablonu

Element	Type	Opis
version	string	(wymagane) Wersja eksportu.
template_groups		(wymagane) Element główny dla grup szablonów.
uuid	string	(wymagane) Unikalny identyfikator tej grupy szablonów.
name	string	(wymagane) Nazwa grupy szablonów.
host_groups		Element główny dla grup hostów używanych przez prototypy hostów.
uuid	string	(wymagane) Unikalny identyfikator tej grupy hostów.
name	string	(wymagane) Nazwa grupy hostów.
templates		Element główny dla szablonów.
triggers		Element główny dla elementów wyzwalaczy szablonu, które są takie same jak dla wyzwalaczy pozycji szablonu.
graphs		Element główny dla elementów wykresów szablonu, które są takie same jak dla wykresów hosta.

Szablony

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      template: 'Linux by Zabbix agent'
      name: 'Linux by Zabbix agent'
      description: |
        To jest oficjalny szablon Linux. Wymaga Zabbix agent 8.0 lub nowszego. (...)
      wizard_ready: 'YES'
      readme: |
        ## Omówienie

        To jest oficjalny szablon Linux. Wymaga Zabbix agent 8.0 lub nowszego. (...)
      vendor:
        name: Zabbix
        version: 8.0-2
      groups:
        - name: 'Templates/Operating systems'
      items: (see table below)
      discovery_rules: (see table below)
      tags:
        - tag: class
          value: os
        - tag: target
          value: linux
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          value: 3m
          description: 'Limit czasu, po którym agent jest uznawany za niedostępny. Działa tylko dla agentó
          config: (see table below)
        - macro: '{$CPU.UTIL.CRIT}'
          value: '90'
          description: 'Krytyczny próg użycia CPU wyrażony w %.'
          config: (see table below)
```

```
      (...)
    dashboards: (see table below)
    valuemaps: (see table below)
  (...)
```

Element	Type	Description
uuid	string	(wymagane) Unikalny identyfikator tego szablonu.
template	string	(wymagane) Unikalna nazwa szablonu.
name	string	Widoczna nazwa szablonu.
description	text	Opis szablonu.
wizard_ready	text	Określa, czy szablon jest dostępny do wyboru w Kreatorze hosta. Możliwe wartości:[1] NO (0, domyślnie), YES (1). Zobacz też: Obiekt Template (`wizard_ready`).
readme	text	Instrukcje konfiguracji specyficzne dla szablonu, wyświetlane w Kreatorze hosta. Obsługuje formatowanie Markdown.
vendor		Element główny dla dostawcy szablonu (obecny, jeśli wyeksportowany szablon zawiera dane dostawcy).
name	string	(wymagane) Nazwa dostawcy szablonu.
version	string	(wymagane) Wersja szablonu. W przypadku szablonów dostarczanych standardowo wersja jest wyświetlana w następującej postaci: główna wersja Zabbix, separator (”-”), numer rewizji (zwiększany przy każdej nowej wersji szablonu i resetowany przy każdej głównej wersji Zabbix). Na przykład 7.0-0, 7.0-3, 8.0-0, 8.0-3.
templates		Element główny dla powiązanych szablonów.
name	string	(wymagane) Nazwa szablonu.
groups		Element główny dla grup szablonów.
name	string	(wymagane) Nazwa grupy szablonów.
items		Element główny dla pozycji szablonu.
discovery_rules		Element główny dla reguł niskopoziomowego wykrywania szablonu.
httptests		Element główny dla scenariuszy web szablonu.
tags		Element główny dla tagów szablonu.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
macros		Element główny dla makr użytkownika szablonu.
macro	string	(wymagane) Nazwa makra użytkownika.
type	string	Typ makra użytkownika. Możliwe wartości:[1] TEXT (0, domyślnie), SECRET_TEXT (1), VAULT (2). Zobacz też: Obiekt User macro (`type`).
value	string	Wartość makra użytkownika.
description	string	Opis makra użytkownika.
config		Element główny dla konfiguracji makra szablonu, odpowiedzialny za sposób wyświetlania makra w Kreatorze hosta.
dashboards		Element główny dla dashboardów szablonu.
valuemaps		Element główny dla map wartości szablonu.

Pozycje szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        - uuid: f94f9f4699e94c369e6c98b2a2f485ce
          name: 'Zabbix agent ping'
          key: agent.ping
          description: 'agent zawsze zwraca „1” dla tej pozycji. Można użyć w połączeniu z `nodata()` do s
          valuemap:
            name: 'Zabbix agent ping status'
          tags:
            - tag: component
              value: system
        (...)
```

```
      - uuid: 58818005e76d46dda14d6592f601ab00
        name: 'Liczba zainstalowanych pakietów'
        key: system.sw.packages.get
        delay: 1h
        preprocessing: (zobacz tabelę poniżej)
        tags:
          - tag: component
            value: os
        triggers: (zobacz tabelę poniżej)
      - uuid: 403cebed115441369e94d35d070ca7b8
        name: 'Wykorzystanie pamięci'
        type: DEPENDENT
        key: vm.memory.utilization
        value_type: FLOAT
        units: '%'
        description: 'Procent wykorzystanej pamięci jest obliczany jako `100-pavailable`.'
        preprocessing: (zobacz tabelę poniżej)
        master_item:
          key: 'vm.memory.size[pavailable]'
        tags:
          - tag: component
            value: memory
        triggers: (zobacz tabelę poniżej)
      (...)
```

Element	Typ	Opis
uuid	string	(wymagane) Unikatowy identyfikator tej pozycji.
name	string	(wymagane) Nazwa pozycji.
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), 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	(wymagane dla pozycji SNMP_AGENT) Identyfikator obiektu SNMP.
key	string	(wymagane) Klucz pozycji.
delay	string	Interwał aktualizacji pozycji. Domyślnie: 1m. Dla pozycji TRAP wartość zawsze wynosi 0.
history	string	Okres (z użyciem sufiksu czasu, makra użytkownika lub makra LLD), przez jaki powinny być przechowywane dane historyczne. Domyślnie: 31d.
trends	string	Okres (z użyciem sufiksu czasu, makra użytkownika lub makra LLD), przez jaki powinny być przechowywane dane trendów. Domyślnie: 365d.
status	string	Stan pozycji. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
value_type	string	Typ odebranej wartości. Możliwe wartości:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, domyślnie), TEXT (4), BINARY (5), JSON (6).
allowed_hosts	string	Lista adresów IP hostów, rozdzielanych przecinkami, które mogą wysyłać dane dla tej pozycji. Obsługiwane dla pozycji TRAP i HTTP_AGENT.
units	string	Jednostki odebranej wartości (bps, B itd.).
params	text	Dodatkowe parametry zależne od typu pozycji (wykonywany skrypt dla pozycji SSH i TELNET; zapytanie SQL dla pozycji ODBC; formuła dla pozycji CALCULATED; skrypt dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER).
ipmi_sensor	string	Czujnik IPMI. Obsługiwane dla pozycji IPMI.

Element	Typ	Opis
authtype	string	Typ uwierzytelniania. Obsługiwane dla pozycji SSH i HTTP_AGENT. Możliwe wartości dla pozycji SSH:[1] PASSWORD (0, domyślnie), PUBLIC_KEY (1). Możliwe wartości dla pozycji HTTP_AGENT:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
username	string	(wymagane dla pozycji SSH i TELNET) Nazwa użytkownika do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `password` (zobacz poniżej) albo pozostawić oba elementy puste.
password	string	(wymagane dla pozycji SSH i TELNET) Hasło do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `username` (zobacz powyżej) albo pozostawić oba elementy puste.
publickey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza publicznego.
privatekey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza prywatnego.
description	text	Opis pozycji.
inventory_link	string	Pole inwentarza hosta wypełniane przez pozycję. Możliwe wartości:[1] NONE (0), ALIAS (4) itd. (obsługiwane pola znajdziesz w sekcji Inwentarz hosta).
valuemap		Element główny map wartości pozycji.
name	string	(wymagane) Nazwa mapy wartości używanej dla pozycji.
logtimefmt	string	Format czasu we wpisach dziennika. Obsługiwane dla pozycji typu wartości LOG.
preprocessing		Element główny wstępnego przetwarzania wartości pozycji.
step		Element główny dla kroków wstępnego przetwarzania wartości pozycji szablonu.
jmx_endpoint	string	Punkt końcowy JMX. Obsługiwane dla pozycji JMX.
master_item		(wymagane dla pozycji DEPENDENT) Element główny pozycji nadrzędnej pozycji zależnej.
key	string	(wymagane) Klucz pozycji nadrzędnej pozycji zależnej.
timeout	string	Limit czasu żądania odpytywania danych pozycji. Obsługiwane dla typów pozycji znajdujących się na liście Limitów czasu.
url	string	(wymagane dla pozycji HTTP_AGENT) Ciąg URL.
query_fields		Element główny parametrów zapytania. Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania. Obsługiwane dla pozycji HTTP_AGENT.
parameters		Element główny parametrów zdefiniowanych przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
name	string	(wymagane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER) Nazwa parametru zdefiniowanego przez użytkownika.
value	string	Wartość parametru zdefiniowanego przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
posts	string	Dane treści żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielane przecinkami. Obsługiwane dla pozycji HTTP_AGENT.
follow_redirects	string	Podążanie za przekierowaniami odpowiedzi podczas odpytywania danych. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
post_type	string	Typ treści danych POST. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (2), XML (3).
http_proxy	string	Ciąg połączenia proxy HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
headers		Element główny nagłówków żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa nagłówka.
value	string	(wymagane dla pozycji HTTP_AGENT) Wartość nagłówka.

Element	Typ	Opis
retrieve_mode	string	Część odpowiedzi, która powinna zostać zapisana. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
request_method	string	Typ metody żądania. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] GET (0, domyślnie), POST (1), PUT (2), HEAD (3).
output_format	string	Sposób przetwarzania odpowiedzi. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (1).
allow_traps	string	Zezwalanie na wypełnianie wartości w sposób podobny do pozycji trappera. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_password	string	Hasło do pliku klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
verify_peer	string	Czy należy sprawdzić, czy certyfikat hosta jest autentyczny. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Czy należy sprawdzić, czy nazwa hosta połączenia odpowiada nazwie w certyfikacie hosta. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
time_shift	string	Jak daleko wstecz (przed bieżącym czasem) kończy się zakres czasu odpytywany przez pozycję. Obsługiwane dla pozycji TELEMETRY_QUERY.
lookback_limit	string	Jak daleko wstecz (przed bieżącym czasem) pozycja odpytuje dane, gdy nie ma poprzedniego zapytania, od którego można kontynuować (np. podczas pierwszego uruchomienia pozycji), lub gdy poprzednie zapytanie jest starsze niż `delay` (np. z powodu niedostępności serwera Zabbix). Obsługiwane dla pozycji TELEMETRY_QUERY.
granularity	string	Długość fragmentu, na który dzielony jest pełny zakres czasu zapytania. Obsługiwane dla pozycji TELEMETRY_QUERY.
query		(wymagane dla pozycji TELEMETRY_QUERY) Element główny zapytania telemetrycznego. Określa, jakie dane pozycja pobiera ze źródła danych, jak je filtruje i jak agreguje wynik.
signal_type	string	(wymagane) Typ sygnału, którego dotyczy zapytanie pozycji. Obsługiwane dla pozycji TELEMETRY_QUERY. Możliwe wartości:[1] TRACES (0), METRICS (1), LOGS (2).
metric_point_type	string	(wymagane dla typu sygnału METRICS) Typ punktu metryki, którego dotyczy zapytanie pozycji. Obsługiwane dla pozycji TELEMETRY_QUERY. Możliwe wartości:[1] SUM (0), GAUGE (1), HISTOGRAM (2), EXPONENTIAL_HISTOGRAM (3).
columns		(wymagane) Element główny kolumn danych telemetrycznych, których dotyczy zapytanie pozycji. Ten element zawiera następujące elementy: `column (string)` - (wymagane) Nazwa kolumny danych telemetrycznych, której dotyczy zapytanie pozycji. `attribute_key (string)` - (wymagane dla kolumn zawierających pary klucz-wartość) Nazwa klucza, którego dotyczy zapytanie pozycji, w kolumnie zawierającej pary klucz-wartość.

Element	Typ	Opis
aggregated_columns		(wymagane) Element główny zagregowanych kolumn danych telemetrycznych, których dotyczy zapytanie pozycji. Ten element zawiera następujące elementy: `column` (`string`) - (wymagane dla funkcji MIN, MAX, AVG, SUM i PERCENTILE) Nazwa kolumny danych telemetrycznych, której dotyczy zapytanie pozycji. `function` (`string`) - (wymagane) Funkcja stosowana przez pozycję do rekordów kolumny. Możliwe wartości:[1] MIN (1), MAX (2), AVG (3), COUNT (4), SUM (5), PERCENTILE (8). `parameters` (`array`) - (wymagane dla funkcji PERCENTILE) Percentyl obliczany przez pozycję na podstawie rekordów kolumny. `alias` (`string`) - (wymagane) Unikatowa nazwa obliczonej wartości, używana do identyfikowania jej w wartości pozycji.
filter		(wymagane) Element główny filtra zapytania telemetrycznego dla rekordów kolumn w zapytaniu. Ten element zawiera następujące elementy: `evaltype` (`string`) - (wymagane) Metoda oceny warunków filtra. Możliwe wartości:[1] AND_OR (0), AND (1), OR (2), FORMULA (3). `formula` (`string`) - (wymagane dla metod oceny FORMULA) Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z niestandardowym wyrażeniem. `conditions` (`string`) - (wymagane) Element główny warunków filtra danych telemetrycznych, które musi spełniać rekord w kolumnie danych telemetrycznych. Element `conditions` zawiera następujące elementy: `column` (`string`) - (wymagane) Nazwa kolumny danych telemetrycznych, do której odnoszą się warunki filtra. `attribute_key` (`string`) - (wymagane dla kolumn zawierających pary klucz-wartość) Nazwa klucza, do którego odnosi się warunek, w kolumnie zawierającej pary klucz-wartość. `operator` (`string`) - (wymagane) Operator warunku filtra. Możliwe wartości dla kolumn zawierających pojedynczą wartość:[1] EQUAL (0), NOT_EQUAL (1), LIKE (2), NOT_LIKE (3). Możliwe wartości dla kolumn zawierających pary klucz-wartość:[1] EQUAL (0), NOT_EQUAL (1), EXISTS (12). `formulaid` (`string`) - (wymagane dla metod oceny FORMULA) Dowolny unikatowy identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. `value` (`string`) - (wymagane dla operacji EQUAL, NOT_EQUAL, LIKE i NOT LIKE) Wartość do porównania.
tags		Element główny tagów pozycji.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
triggers		Element główny dla wyzwalaczy pozycji szablonu.

Note:
Zobacz także: Obiekt pozycji (odpowiednią właściwość o pasującej nazwie).

Kroki przetwarzania wstępnego wartości pozycja szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          preprocessing:
            - type: JSONPATH
              parameters:
                - $.length()
```

```
          - type: DISCARD_UNCHANGED_HEARTBEAT
            parameters:
              - 12h
        (...)
```

Element	Type	Description
type	string	(wymagane) Typ kroku przetwarzania wstępnego wartości pozycji. Dostępne wartości:[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		(wymagane) Element główny dla parametrów kroku przetwarzania wstępnego wartości pozycji.
parameter	string	Pojedynczy parametr kroku przetwarzania wstępnego wartości pozycji.
error_handler	string	Typ akcji używany w przypadku niepowodzenia kroku przetwarzania wstępnego. Dostępne wartości:[1] ORIGINAL_ERROR (0, domyślnie), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3).
error_handler_params	string	Parametry obsługi błędów.

Note:
Zobacz też: Obiekt przetwarzania wstępnego pozycji (odwołanie do odpowiedniej właściwości o zgodnej nazwie).

Wyzwalacze pozycji szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      items:
        (...)
        - uuid: 58818005e76d46dda14d6592f601ab00
          (...)
          triggers:
            - uuid: b950c306394f4b3c902060a8273cbcde
              expression: 'change(/Linux by Zabbix agent/system.sw.packages.get)<>0'
              name: 'Linux: Liczba zainstalowanych pakietów została zmieniona'
              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)>{$MEMORY.UTIL.MAX}'
              name: 'Linux: Wysokie wykorzystanie pamięci'
              event_name: 'Linux: Wysokie wykorzystanie pamięci (>{$MEMORY.UTIL.MAX}% przez 5m)'
              priority: AVERAGE
              description: 'Systemowi zaczyna brakować wolnej pamięci.'
              dependencies:
                - name: 'Linux: Brak dostępnej pamięci'
                  expression: 'max(/Linux by Zabbix agent/vm.memory.size[available],5m)<{$MEMORY.AVAILABLE
              tags:
```

```
            - tag: scope
              value: capacity
            - tag: scope
              value: performance
      (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego wyzwalacza.
expression	string	(wymagane) Wyrażenie wyzwalacza.
recovery_mode	string	Podstawa generowania zdarzeń OK. Możliwe wartości:[1] EXPRESSION (0, domyślnie), RECOVERY_EXPRESSION (1), NONE (2).
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza.
correlation_mode	string	Tryb korelacji (bez korelacji zdarzeń lub korelacja zdarzeń według tagu). Możliwe wartości:[1] DISABLED (0, domyślnie), TAG_VALUE (1).
correlation_tag	string	Nazwa tagu używana do korelacji zdarzeń.
name	string	(wymagane) Nazwa wyzwalacza.
event_name	string	Nazwa zdarzenia.
opdata	string	Dane operacyjne.
url_name	string	Etykieta dla adresu URL powiązanego z wyzwalaczem.
url	string	Adres URL powiązany z wyzwalaczem.
status	string	Status wyzwalacza. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
priority	string	Ważność wyzwalacza. Możliwe wartości:[1] NOT_CLASSIFIED (0, domyślnie), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5).
description	text	Opis wyzwalacza.
type	string	Typ generowania zdarzeń (pojedyncze zdarzenie problemu lub wiele zdarzeń problemu). Możliwe wartości:[1] SINGLE (0, domyślnie), MULTIPLE (1).
manual_close	string	Ręczne zamykanie zdarzeń problemu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
dependencies		Element główny dla zależności.
name	string	(wymagane) Nazwa wyzwalacza zależności.
expression	string	(wymagane) Wyrażenie wyzwalacza zależności.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza zależności.
tags		Element główny dla tagów wyzwalacza.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.

Note:
Zobacz też: Trigger object (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Reguły wykrywania niskiego poziomu szablonu

```
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:
Większość elementów reguły wykrywania niskiego poziomu szablonu jest taka sama jak w przypadku pozycji szablonu. Poniższa tabela opisuje te elementy, które różnią się od pozycji szablonu.

Element	Type	Description
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), 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	(wymagane) Klucz reguły wykrywania niskiego poziomu.
filter		Element główny dla filtrów reguły wykrywania niskiego poziomu szablonu.
lifetime	string	Okres czasu (w sekundach, z przyrostkiem czasu lub makrem użytkownika), po którym zasoby, które nie są już wykrywane, zostaną usunięte. Domyślnie: 7d.
lifetime_type	string	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER.
enabled_lifetime	string	Okres czasu (w sekundach, z przyrostkiem czasu lub makrem użytkownika), po którym zasoby, które nie są już wykrywane, zostaną wyłączone.
enabled_lifetime_type	string	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER.
item_prototypes		Element główny dla elementów prototypów pozycji szablonu, które są takie same jak w przypadku pozycji szablonu.
trigger_prototypes		Element główny dla elementów prototypów wyzwalaczy szablonu, które są takie same jak w przypadku wyzwalaczy pozycji szablonu.
graph_prototypes		Element główny dla elementów prototypów wykresów szablonu, które są takie same jak w przypadku wykresów hosta.
host_prototypes		Element główny dla elementów prototypów hosta szablonu, które są takie same jak w przypadku hostów.
parent_discovery_rule		Element główny dla nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły) prototypu reguły wykrywania niskiego poziomu. Ta właściwość oznacza, że jest to prototyp reguły LLD, bezpośredni element podrzędny wskazanej reguły/prototypu reguły.
key	string	(wymagane) Klucz nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły).
master_item	string	(wymagane dla reguł DEPENDENT) Element główny dla elementu nadrzędnego reguły zależnej.
lld_macro_paths		Element główny dla ścieżek makr reguły wykrywania niskiego poziomu.
lld_macro	string	(wymagane) Nazwa makra reguły wykrywania niskiego poziomu.
path	string	(wymagane) Selektor wartości, która zostanie przypisana do odpowiedniego makra.
preprocessing		Element główny dla wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu.
step		Element główny dla elementów kroków wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu, które są takie same jak w przypadku kroków wstępnego przetwarzania wartości pozycji szablonu, z wyjątkiem mniejszej liczby możliwych wartości. Zobacz też: obiekt wstępnego przetwarzania reguły LLD.
overrides		Element główny dla reguł nadpisywania reguły wykrywania niskiego poziomu.
name	string	(wymagane) Unikalna nazwa nadpisania.
step	string	(wymagane) Unikalny numer kolejności nadpisania.
stop	string	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli pasuje.
filter		Element główny dla elementów filtra reguły nadpisywania reguły wykrywania niskiego poziomu szablonu, które są takie same jak w przypadku filtrów reguły wykrywania niskiego poziomu szablonu.
operations		Element główny dla operacji nadpisywania reguły wykrywania niskiego poziomu szablonu.

Note:
Zobacz też: obiekt reguły LLD (odwołaj się do odpowiedniej właściwości o zgodnej nazwie).

Filtry reguły wykrywania niskiego poziomu szablonu

```
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	Typ	Opis
evaltype	string	Zastępuje metodę oceny warunków filtra. Dostępne wartości:[1] AND_OR (0, domyślnie), AND (1), OR (2), FORMULA (3).
formula	string	Niestandardowa formuła obliczeniowa dla warunków filtra.
conditions		Element główny dla warunków filtra.
macro	string	(wymagane) Nazwa makra wykrywania niskiego poziomu, na którym ma zostać wykonane sprawdzenie.
value	string	Wartość do porównania.
operator	string	Operator warunku. Dostępne wartości:[1] MATCHES_REGEX (8, domyślnie), NOT_MATCHES_REGEX (9).
formulaid	string	(wymagane) Dowolny unikalny identyfikator używany do odwołania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale przy kolejnym pobraniu zostanie wygenerowany ponownie.

Note:
Zobacz także: Obiekt filtra reguły LLD (odwołanie do odpowiedniej właściwości o zgodnej nazwie).

Operacje nadpisania reguły wykrywania niskiego poziomu szablonu

Element	Type	Description
operationobject	string	Obiekt, do którego jest stosowana operacja. Możliwe wartości:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3).
operator	string	Operator warunku nadpisania. Możliwe wartości:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6).
value	string	Wyrażenie regularne lub ciąg znaków dla operatora warunku nadpisania.
status	string	Status obiektu po operacji nadpisania.
discover	string	Czy obiekt jest dodawany w wyniku wykrycia.

Element	Type	Description
delay	string	Interwał aktualizacji ustawiony dla prototypu pozycji po operacji nadpisania.
history	string	Okres przechowywania historii ustawiony dla prototypu pozycji po operacji nadpisania.
trends	string	Okres przechowywania trendów ustawiony dla prototypu pozycji po operacji nadpisania.
severity	string	Poziom ważności prototypu wyzwalacza ustawiony po operacji nadpisania.
tags		Element główny dla tagów ustawionych dla obiektu po operacji nadpisania.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
templates		Element główny dla szablonów powiązanych z prototypem hosta po operacji nadpisania.
name	string	(wymagane) Nazwa szablonu.
inventory_mode	string	Tryb inwentarza prototypu hosta ustawiony po operacji nadpisania.

Note:
Zobacz też: Obiekt operacji nadpisania reguły LLD (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Szablon scenariuszy webowych

Element	Type	Description
uuid	string	(wymagane) Unikalny identyfikator tego scenariusza webowego.
name	string	(wymagane) Nazwa scenariusza webowego.
delay	string	Częstotliwość (w sekundach, przyrostek czasu lub makro użytkownika) wykonywania scenariusza webowego. Domyślnie: 1m.
attempts	integer	Liczba prób wykonania kroków scenariusza webowego. Możliwe wartości:[1] 1-10 (domyślnie: 1).
agent	string	Agent klienta. Zabbix będzie podszywać się pod wybraną przeglądarkę. Jest to przydatne, gdy witryna zwraca różną zawartość dla różnych przeglądarek. Domyślnie: Zabbix.
http_proxy	string	Proxy używany przez scenariusz webowy, podawany jako: `http://[username[:password]@]proxy.example.com[:port]`
variables		Element główny dla zmiennych scenariusza webowego, które mogą być używane w krokach scenariusza.
name	string	(wymagane) Nazwa zmiennej.
value	text	(wymagane) Wartość zmiennej.
headers		Element główny dla nagłówków HTTP wysyłanych podczas wykonywania żądania. Nagłówki powinny być wymienione przy użyciu tej samej składni, w jakiej występowałyby w protokole HTTP.
name	string	(wymagane) Nazwa nagłówka.
value	text	(wymagane) Wartość nagłówka.
status	string	Status scenariusza webowego. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
authentication	string	Metoda uwierzytelniania. Możliwe wartości:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
http_user	string	Nazwa użytkownika używana do uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
http_password	string	Hasło używane do uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
verify_peer	string	Sprawdź certyfikat SSL serwera webowego. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Sprawdź, czy pole *Common Name* lub pole *Subject Alternate Name* certyfikatu serwera webowego jest zgodne. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_file	string	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_password	string	Hasło pliku prywatnego klucza SSL.
steps		(wymagane) Element główny dla kroków scenariusza webowego szablonu.
tags		Element główny dla tagów scenariusza webowego.
tag	string	(wymagane) Nazwa tagu.

Element	Type	Description
value	string	Wartość tagu.

Note:
Zobacz także: Obiekt scenariusza webowego (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Kroki scenariusza web szablonu

Element	Type	Description
name	string	(required) Nazwa kroku scenariusza web.
url	string	(required) URL do monitorowania.
query_fields		Element główny dla parametrów zapytania (tablica pól HTTP, które mają zostać dodane do URL podczas wykonywania żądania).
name	string	(required) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania.
posts		Element główny dla zmiennych HTTP POST (ciąg znaków (surowe dane POST) lub tablica pól HTTP (dane pól formularza)).
name	string	(required) Nazwa pola POST.
value	string	(required) Wartość pola POST.
variables		Element główny zmiennych na poziomie kroku (makr), które powinny zostać zastosowane po tym kroku. Jeśli wartość zmiennej ma prefiks 'regex:', jej wartość jest wyodrębniana z danych zwróconych przez ten krok zgodnie z wyrażeniem regularnym następującym po prefiksie 'regex:'
name	string	(required) Nazwa zmiennej.
value	text	(required) Wartość zmiennej.
headers		Element główny nagłówków HTTP, które mają zostać wysłane podczas wykonywania żądania.
name	string	(required) Nazwa nagłówka.
value	text	(required) Wartość nagłówka.
follow_redirects	string	Podążaj za przekierowaniami HTTP. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
retrieve_mode	string	Tryb pobierania odpowiedzi HTTP. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
timeout	string	Limit czasu (w sekundach, time suffix lub user macro) wykonania kroku. Domyślnie: 15s.
required	string	Tekst, który musi być obecny w odpowiedzi (ignorowany, jeśli pusty).
status_codes	string	Lista kodów statusu HTTP rozdzielona przecinkami, które są akceptowane (np. 200-201,210-299; ignorowane, jeśli puste).

Note:
Zobacz także: Obiekt kroku scenariusza web (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Konfiguracja makra szablonu

```
zabbix_export:
  (...)
  templates:
    - uuid: f8f7908280354f2abeed07dc788c3747
      (...)
      macros:
        - macro: '{$AGENT.TIMEOUT}'
          (...)
          config:
            type: TEXT
            priority: '1'
            label: 'Sekundy od ostatniego wykrycia agenta Zabbix'
            description: 'Limit czasu, po którym agent jest uznawany za niedostępny.'
        - macro: '{$CPU.UTIL.CRIT}'
```

```
      (...)
      config:
        type: TEXT
        priority: '2'
        section_name: Thresholds
        label: 'Próg wykorzystania CPU wyrażony'
        description: 'Krytyczny próg wykorzystania CPU wyrażony w %. W zakresie od 0 do 100 włącznie.'
        regex: '^-?([0-9]+|(([0-9]+)\.([0-9]+)))$'
    (...)
    - macro: '{$IFCONTROL}'
      (...)
      config:
        type: CHECKBOX
        priority: '19'
        label: 'Kontrola interfejsów'
        description: 'Wyzwól wyzwalacz, gdy status operacyjny interfejsu zmieni się na "Link down".'
        options:
          - checked: '1'
            unchecked: '0'
    (...)
```

Element	Type	Description
type	string	(wymagane) Typ pola wejściowego makra. Możliwe wartości:[1] NOCONF (0), TEXT (1), LIST (2), CHECKBOX (3).
priority	string	Pozycja makra na liście makr.
section_name	string	Etykieta zwijanej sekcji, w której grupowane jest makro.
label	string	(wymagane dla makr TEXT, LIST, CHECKBOX) Etykieta makra.
description	text	Tekst pomocy dla makra. Obsługuje formatowanie Markdown.
required	string	Oznacza makro jako obowiązkowe. Możliwe wartości:[1] NO (0), YES (1). Obsługiwane dla makr TEXT i LIST.
regex	string	Wyrażenie regularne do walidacji danych wprowadzonych przez użytkownika w polu tekstowym. Obsługiwane dla makr TEXT.
options		Element główny dla pozycji LIST lub wartości CHECKBOX.
value	string	(wymagane dla makr LIST) Wartość pozycji LIST.
text	string	(wymagane dla makr LIST) Tekst pozycji LIST.
checked	string	(wymagane dla makr CHECKBOX) Wartość reprezentująca stan zaznaczony. Możliwe wartości:[1] 0 (false), 1 (true). Obsługiwane dla makr CHECKBOX.
unchecked	string	(wymagane dla makr CHECKBOX) Wartość reprezentująca stan niezaznaczony. Możliwe wartości:[1] 0 (false), 1 (true). Obsługiwane dla makr CHECKBOX.

Note:
Zobacz także: Obiekt konfiguracji makra (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Pulpity szablonu

```
zabbix_export:
  (...)
  templates:
    (...)
      dashboards:
        - uuid: c689ad3115fd46a4b927d1f70ee2e5a4
          name: Filesystems
          pages:
            - name: Overview
              widgets: (see table below)
        (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tego pulpitu.
name	string	(wymagane) Nazwa pulpitu szablonu.
display pe- riod	integer	Okres wyświetlania stron pulpitu.
auto_start	string	Automatyczne uruchamianie pokazu slajdów. Possible values:[1] NO (0), YES (1, default).
pages		Element główny dla stron pulpitu szablonu.
name	string	Nazwa strony.
display period	integer	Interwał pokazu slajdów.
sortorder	integer	Kolejność sortowania strony.
widgets		Element główny dla widżetów pulpitu szablonu.

Note:
Zobacz także: Obiekt pulpitu szablonu (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Widżety pulpitu szablonu

```
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})]: Wykres wykorzystania przestrzeni (względem całoś
                    - 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})]: Wykres użycia przestrzeni, w % (względem maksyma
                    - type: STRING
                      name: reference
                      value: AAAEC
        (...)
```

Element	Type	Opis
type	string	(wymagane) Typ widżetu.
name	string	Nazwa widżetu.
x	integer	Pozycja pozioma od lewej strony pulpitu szablonu. Możliwe wartości:[1] 0-71.
y	integer	Pozycja pionowa od górnej strony pulpitu szablonu. Możliwe wartości:[1] 0-63.
width	integer	Szerokość widżetu. Możliwe wartości:[1] 1-72.
height	integer	Wysokość widżetu. Możliwe wartości:[1] 1-64.
hide_header	string	Ukryj nagłówek widżetu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
fields		Element główny dla pól widżetu pulpitu szablonu.
type	string	(wymagane) Typ pola widżetu. Możliwe wartości:[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	(wymagane) Nazwa pola widżetu.
value	mixed	(wymagane) Wartość pola widżetu, zależnie od typu pola.

Note:
Zobacz także: Obiekt widżetu pulpitu szablonu (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Mapy wartości szablonu

```
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	Opis
uuid	string	(wymagane) Unikalny identyfikator tej mapy wartości.
name	string	(wymagane) Nazwa mapy wartości.
mapping		Element główny dla mapowań.
type	string	Typ dopasowania mapowania. Możliwe wartości:[1] EQUAL (0, domyślnie), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6).
value	string	Wartość oryginalna.

Element	Type	Opis
newvalue	string	(wymagane) Wartość, na którą mapowana jest wartość oryginalna.

Note:
Zobacz także: Obiekt mapy wartości (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Przypisy [1] Wartości całkowite API w nawiasach, na przykład ENABLED (0), są podane wyłącznie jako informacja pomocnicza. Więcej informacji można znaleźć na powiązanej stronie obiektu API w wpisie tabeli lub na końcu każdej sekcji.

Hosty

Przegląd

Hosty są eksportowane wraz z wieloma powiązanymi obiektami i relacjami między obiektami.

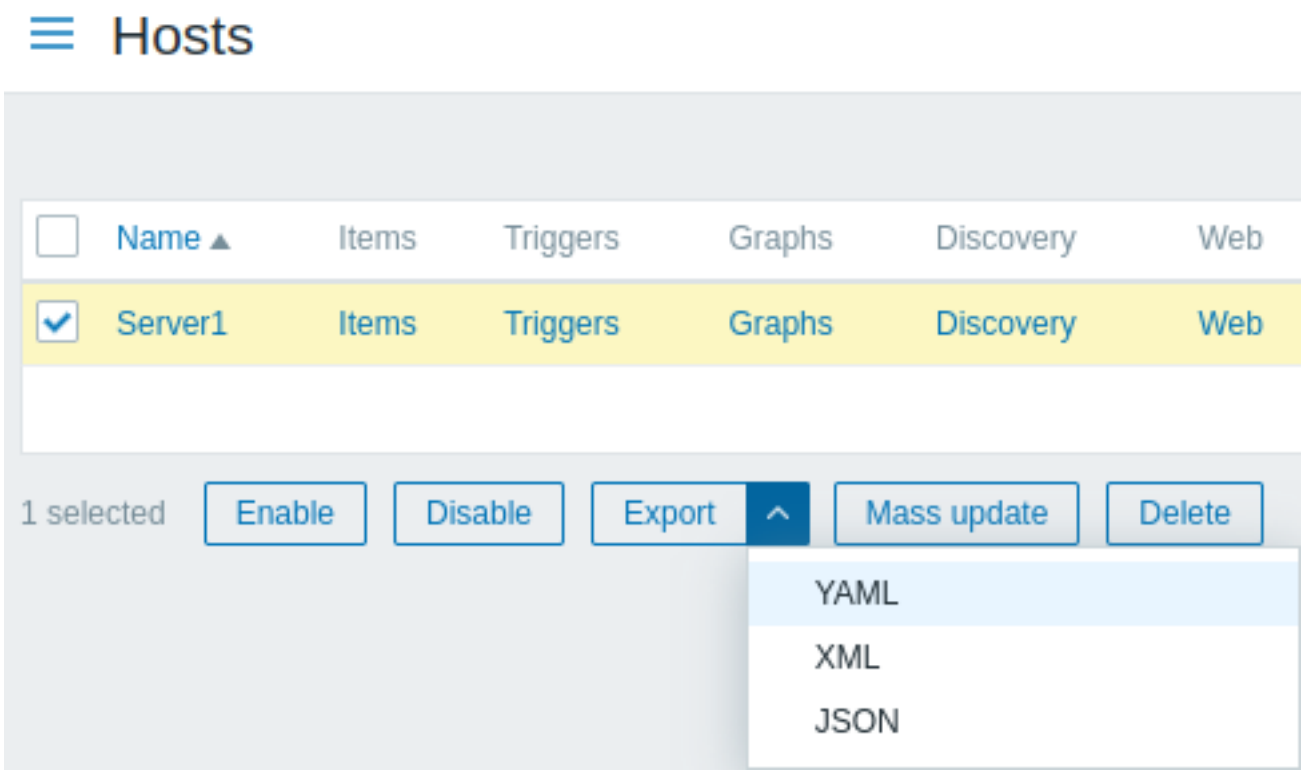
Eksport hosta zawiera:

- Powiązane grupy hostów
- Dane hosta
- Powiązania z szablonami
- Powiązania z grupami hostów
- Interfejsy hosta
- Bezpośrednio powiązane pozycje
- Bezpośrednio powiązane wyzwalacze
- Bezpośrednio powiązane reguły wykrywania wraz ze wszystkimi prototypami
- Bezpośrednio powiązane scenariusze web
- Makra hosta
- Dane inwentarza hosta
- Mapowania wartości
- Powiązane wykresy

Eksportowanie

Aby wyeksportować hosty, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Hosts*.
2. Zaznacz pola wyboru hostów do wyeksportowania.
3. Kliknij *Export* pod listą, aby wyeksportować w domyślnym formacie YAML, lub kliknij strzałkę, aby wybrać inny format eksportu.



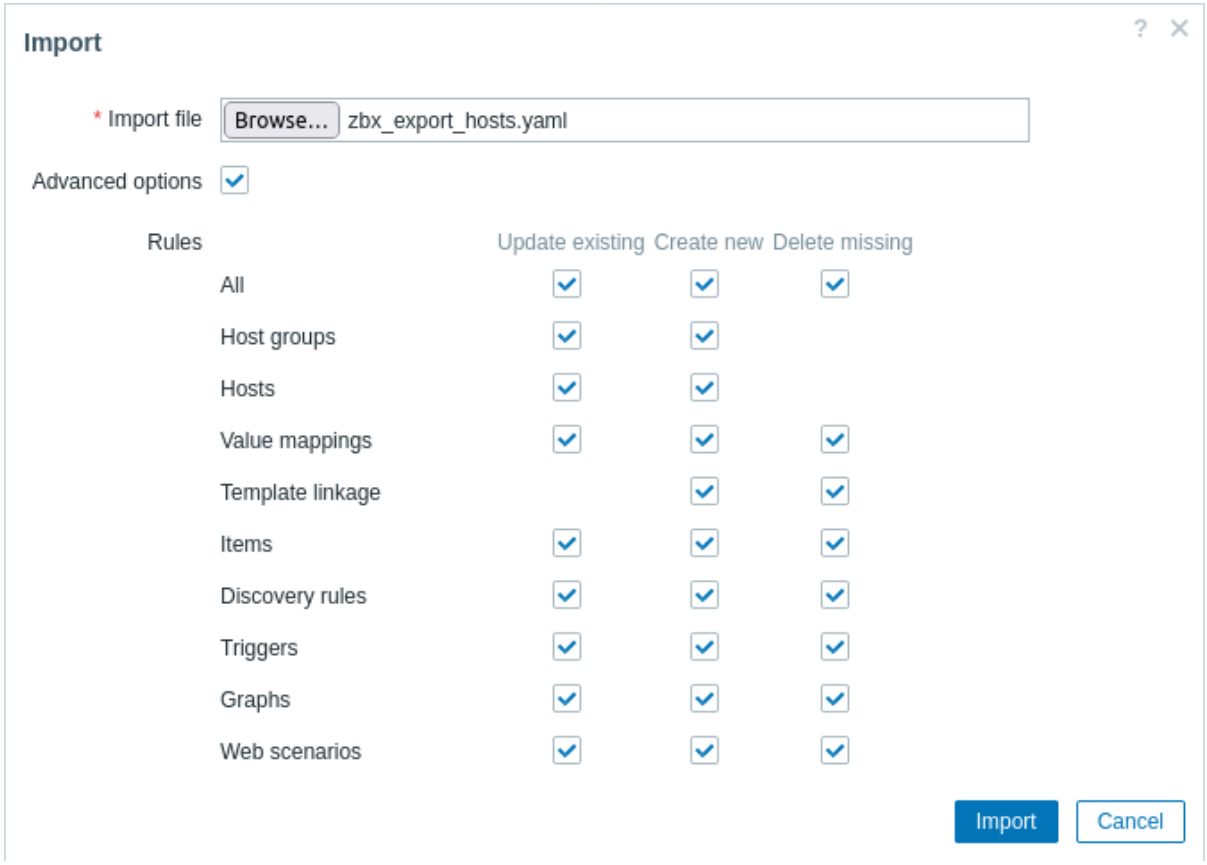
W zależności od wybranego formatu hosty są eksportowane do lokalnego pliku o domyślnej nazwie:

- `zabbix_export_hosts.yaml` - w eksporcie YAML;
- `zabbix_export_hosts.xml` - w eksporcie XML;
- `zabbix_export_hosts.json` - w eksporcie JSON.

Importowanie

Aby zaimportować hosty, wykonaj następujące czynności:

1. Przejdź do sekcji *Data collection* > *Hosts*.
2. Kliknij przycisk *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz pola wyboru reguł importu, które chcesz zastosować (zobacz poniżej).
5. Kliknij przycisk *Import* w prawym dolnym rogu formularza konfiguracji.



Jeśli zaznaczysz pole wyboru *Advanced options*, zostanie wyświetlona szczegółowa lista wszystkich elementów, które można zaimportować — zaznacz lub odznacz poszczególne reguły importu zgodnie z wymaganiami.

Jeśli klikniesz pole wyboru w wierszu *All*, wszystkie znajdujące się poniżej elementy zostaną zaznaczone lub odznaczone.

Reguły importu:

Rule	Description
Update existing	Istniejące elementy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe elementy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.
Delete missing	Istniejące elementy, których nie ma w pliku importu, zostaną usunięte. W przeciwnym razie nie zostaną usunięte. Jeśli opcja *Delete missing* jest zaznaczona dla *Template linkage*, istniejące powiązania z szablonami, których nie ma w pliku importu, zostaną usunięte. Encje (pozycje, wyzwalacze, wykresy itd.) odziedziczone z odłączonych szablonów nie zostaną usunięte (chyba że opcja *Delete missing* zostanie również wybrana dla każdej encji).

Podczas importu nie można przejrzeć zmian dotyczących hostów. Komunikat o pomyślnym lub nieudanym imporcie zostanie wyświetlony w frontendzie.

Format eksportu

Gdy host jest eksportowany, Zabbix generuje uporządkowany format YAML, JSON lub XML. Eksport obejmuje elementy hosta, takie jak metadane hosta, pozycje, makra, wyzwalacze i inne.

Każdy element pełni określoną rolę i może zawierać elementy zagnieżdżone.

W poniższych sekcjach opisano każdy element w formacie eksportu. W przykładach użyto hosta z szablonem MySQL by Zabbix agent 2. Na potrzeby demonstracji elementów szablon jest odłączany po utworzeniu hosta. Odłączenie zachowuje wszystkie elementy w konfiguracji (w przeciwieństwie do odłączenia i wyczyszczenia, które usuwa je).

Wielokropek (`...`) oznacza elementy pominięte ze względu na zwięzłość. Uwaga (`see table below`) jest używana, gdy element jest wyjaśniany bardziej szczegółowo w późniejszych sekcjach.

```
zabbix_export:
  version: '8.0'
  host_groups:
    - uuid: 748ad4d098d447d492bb935c907f652f
      name: Databases
  hosts: (patrz tabela poniżej)
  graphs: (patrz tabela poniżej)
```

Metadane hosta

Element		Type	Description
version		string	(wymagane) Wersja Zabbix.
host_groups			(wymagane) Element główny dla grup hostów.
	uuid	string	(wymagane) Unikalny identyfikator tej grupy hostów.
	name	string	(wymagane) Nazwa grupy hostów.
hosts			Element główny dla hostów.
graphs			Element główny dla wykresów hosta.

Hosts

```
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: 'Hasło użytkownika MySQL.'
        (...)
      valuemaps: (see table below)
  (...)
```

Element	Type	Opis
host	string	(wymagane) Unikalna nazwa hosta.
name	string	Widoczna nazwa hosta.
description	text	Opis hosta.
monitored_by	string	Sposób monitorowania hosta. Dopuszczalne wartości:[1] SERVER (0, domyślnie), PROXY (1) lub PROXY_GROUP (2).
proxy		Element główny dla proxy.
name	string	(wymagane) Nazwa proxy (jeśli dotyczy), które monitoruje host.
proxy_group		Element główny dla grupy proxy.
name	string	(wymagane) Nazwa grupy proxy (jeśli dotyczy), używanej do monitorowania hosta.
status	string	Status hosta. Dopuszczalne wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
ipmi_authtype	string	Typ uwierzytelniania sesji IPMI. Dopuszczalne wartości:[1] DEFAULT (-1, domyślnie), NONE (0), MD2 (1), MD5 (2), STRAIGHT (4), OEM (5), RMCP_PLUS (6).

Element	Type	Opis
ipmi_privilege	string	Poziom uprawnień sesji IPMI. Dopuszczalne wartości:[1] CALLBACK (1), USER (2, domyślnie), OPERATOR (3), ADMIN (4), OEM (5).
ipmi_username	string	Nazwa użytkownika do sprawdzeń IPMI.
ipmi_password	string	Hasło do sprawdzeń IPMI.
templates		Element główny dla powiązanych szablonów.
name	string	(wymagane) Nazwa szablonu.
groups		Element główny dla grup hostów, do których należy host.
name	string	(wymagane) Nazwa grupy hostów.
interfaces		Element główny dla interfejsów hosta.
items		Element główny dla pozycji hosta.
discovery_rules		Element główny dla reguł niskopoziomowego wykrywania hosta.
http_tests		Element główny dla scenariuszy web hosta.
tags		Element główny dla tagów hosta.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
macros		Element główny dla makr hosta.
macro	string	(wymagane) Nazwa makra użytkownika.
type	string	Typ makra użytkownika. Dopuszczalne wartości:[1] TEXT (0, domyślnie), SECRET_TEXT (1), VAULT (2).
value	string	Wartość makra użytkownika.
description	string	Opis makra użytkownika.
inventory		Element główny dla inwentarza hosta.
<inventory_property>	string	Właściwość inwentarza. Wszystkie właściwości mają odpowiadające im elementy (`type`, `name`, `os` itd.; na przykład zobacz Format eksportu).
inventory_mode	string	Tryb inwentarza. Dopuszczalne wartości:[1] DISABLED (-1), MANUAL (0, domyślnie), AUTOMATIC (1).
valuemaps		Element główny dla map wartości hosta.

Note:
Zobacz także: Obiekt Host (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Interfejsy hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      interfaces:
        - ip: 192.0.2.1
          interface_ref: if1
      (...)
```

Element	Type	Opis
default	string	Czy jest to główny interfejs hosta. Należy pamiętać, że na hoście może istnieć tylko jeden główny interfejs danego typu. Dopuszczalne wartości:[1] NO (0), YES (1, domyślnie).
type	string	Typ interfejsu. Dopuszczalne wartości:[1] ZABBIX (1, domyślnie), SNMP (2), IPMI (3), JMX (4).
useip	string	Czy używać adresu IP jako interfejsu do łączenia z hostem (w przeciwnym razie zostanie użyty DNS). Dopuszczalne wartości:[1] NO (0), YES (1, domyślnie).
ip	string	(wymagane dla połączeń IP) Adres IP (IPv4 lub IPv6).
dns	string	(wymagane dla połączeń DNS) Nazwa DNS.
port	string	Numer portu.
details		Element główny dla szczegółów interfejsu.
version	string	Użyj tej wersji SNMP. Dopuszczalne wartości:[1] SNMPV1 (1), SNMP_V2C (2, domyślnie), SNMP_V3 (3).
community	string	(wymagane dla pozycji SNMPv1 i SNMPv2) Społeczność SNMP.

Element	Type	Opis
max_repetitions	string	Maksymalna wartość powtórzeń dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs). Obsługiwane dla pozycji SNMPv2 i SNMPv3 (`discovery[]` i `walk[]`). Domyślnie: 10.
contextname	string	Nazwa kontekstu SNMPv3. Obsługiwane dla pozycji SNMPv3.
securityname	string	Nazwa zabezpieczeń SNMPv3. Obsługiwane dla pozycji SNMPv3.
securitylevel	string	Poziom zabezpieczeń SNMPv3. Obsługiwane dla pozycji SNMPv3. Dopuszczalne wartości:[1] NOAUTHNOPRIV (0, domyślnie), AUTHNOPRIV (1), AUTHPRIV (2).
authprotocol	string	Protokół uwierzytelniania SNMPv3. Obsługiwane dla pozycji SNMPv3. Dopuszczalne wartości:[1] MD5 (0, domyślnie), SHA1 (1), SHA224 (2), SHA256 (3), SHA384 (4), SHA512 (5).
authpassphrase	string	Hasło uwierzytelniania SNMPv3. Obsługiwane dla pozycji SNMPv3.
privprotocol	string	Protokół prywatności SNMPv3. Obsługiwane dla pozycji SNMPv3. Dopuszczalne wartości:[1] DES (0, domyślnie), AES128 (1), AES192 (2), AES256 (3), AES192C (4), AES256C (5).
privpassphrase	string	Hasło prywatności SNMPv3. Obsługiwane dla pozycji SNMPv3.
bulk	string	Używaj żądań zbiorczych dla SNMP. Dopuszczalne wartości:[1] NO (0), YES (1, domyślnie).
interface_ref	string	Nazwa referencyjna interfejsu, która ma być używana w pozycjach (format: `if<N>`).

Note:
Zobacz też: Obiekt interfejsu hosta (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Pozycje hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: 'Wykorzystanie pamięci podręcznej binlog na dysku'
          type: DEPENDENT
          key: mysql.binlog_cache_disk_use
          value_type: FLOAT
          description: 'Liczba transakcji, które użyły tymczasowej pamięci podręcznej na dysku, ponieważ n
          preprocessing: (zobacz tabelę poniżej)
          master_item:
            key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          tags:
            - tag: component
              value: cache
        (...)
        - name: 'Wykorzystanie puli buforów'
          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: 'Stosunek liczby używanych stron do całkowitej liczby stron w puli buforów.'
        tags:
          - tag: component
            value: memory
        triggers: (zobacz tabelę poniżej)
      (...)
      - name: Czas działania
        type: DEPENDENT
        key: mysql.uptime
        units: uptime
        description: 'Liczba sekund, przez które serwer działał.'
        preprocessing: (zobacz tabelę poniżej)
        master_item:
          key: 'mysql.get_status_variables["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
        tags:
          - tag: component
            value: application
        triggers: (zobacz tabelę poniżej)
      (...)
```

Element	Typ	Opis
name	string	(wymagane) Nazwa pozycji.
type	string	Typ pozycji. Możliwe wartości:[1] ZABBIX_PASSIVE (0, domyślnie), 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	(wymagane dla pozycji SNMP_AGENT) Identyfikator obiektu SNMP.
key	string	(wymagane) Klucz pozycji.
delay	string	Interwał aktualizacji pozycji. Domyślnie: 1m. Dla pozycji TRAP wartość zawsze wynosi 0.
history	string	Okres (z użyciem przyrostka czasu, makra użytkownika lub makra LLD), przez który powinny być przechowywane dane historyczne. Domyślnie: 31d.
trends	string	Okres (z użyciem przyrostka czasu, makra użytkownika lub makra LLD), przez który powinny być przechowywane dane trendów. Domyślnie: 365d.
status	string	Stan pozycji. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
value_type	string	Typ odebranej wartości. Możliwe wartości:[1] FLOAT (0), CHAR (1), LOG (2), UNSIGNED (3, domyślnie), TEXT (4), BINARY (5), JSON (6).
allowed_hosts	string	Lista rozdzielanych przecinkami adresów IP hostów, które mogą wysyłać dane dla pozycji. Obsługiwane dla pozycji TRAP i HTTP_AGENT.
units	string	Jednostki odebranej wartości (bps, B itd.).
params	text	Dodatkowe parametry zależne od typu pozycji (wykonywany skrypt dla pozycji SSH i TELNET; zapytanie SQL dla pozycji ODBC; formuła dla pozycji CALCULATED; skrypt dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER).
ipmi_sensor	string	Czujnik IPMI. Obsługiwane dla pozycji IPMI.
authtype	string	Typ uwierzytelniania. Obsługiwane dla pozycji SSH i HTTP_AGENT. Możliwe wartości dla pozycji SSH:[1] PASSWORD (0, domyślnie), PUBLIC_KEY (1). Możliwe wartości dla pozycji HTTP_AGENT:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).

Element	Typ	Opis
username	string	(wymagane dla pozycji SSH i TELNET) Nazwa użytkownika do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `password` (zobacz poniżej) albo pozostawić oba elementy puste.
password	string	(wymagane dla pozycji SSH i TELNET) Hasło do uwierzytelniania. Obsługiwane dla pozycji SIMPLE, ODBC, JMX i HTTP_AGENT. W przypadku użycia dla pozycji JMX należy również określić `username` (zobacz powyżej) albo pozostawić oba elementy puste.
publickey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza publicznego.
privatekey	string	(wymagane dla pozycji SSH) Nazwa pliku klucza prywatnego.
description	text	Opis pozycji.
inventory_link	string	Pole inwentarza hosta, które jest wypełniane przez pozycję. Możliwe wartości:[1] NONE (0), ALIAS (4) itd. (zobacz Inwentarz hosta, aby sprawdzić obsługiwane pola).
valuemap		Element główny map wartości pozycji.
name	string	(wymagane) Nazwa mapy wartości używanej dla pozycji.
logtimefmt	string	Format czasu we wpisach dziennika. Obsługiwane dla pozycji typu wartości LOG.
preprocessing		Element główny wstępnego przetwarzania wartości pozycji.
step		Element główny kroków wstępnego przetwarzania wartości pozycji hosta.
interface_ref	string	Odwołanie do interfejsu hosta (format: `if<N>`).
jmx_endpoint	string	Punkt końcowy JMX. Obsługiwane dla pozycji JMX.
master_item		(wymagane dla pozycji DEPENDENT) Element główny nadrzędnej pozycji pozycji zależnej.
key	string	(wymagane) Klucz nadrzędnej pozycji pozycji zależnej.
timeout	string	Limit czasu żądania odpytywania danych pozycji. Obsługiwane dla listy limitów czasu typów pozycji.
url	string	(wymagane dla pozycji HTTP_AGENT) Ciąg URL.
query_fields		Element główny parametrów zapytania. Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania. Obsługiwane dla pozycji HTTP_AGENT.
parameters		Element główny parametrów zdefiniowanych przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
name	string	(wymagane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER) Nazwa parametru zdefiniowanego przez użytkownika.
value	string	Wartość parametru zdefiniowanego przez użytkownika. Obsługiwane dla pozycji ITEM_TYPE_SCRIPT i ITEM_TYPE_BROWSER.
posts	string	Dane treści żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielane przecinkami. Obsługiwane dla pozycji HTTP_AGENT.
follow_redirects	string	Podążanie za przekierowaniami odpowiedzi podczas odpytywania danych. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
post_type	string	Typ treści danych wysyłanych w żądaniu. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (2), XML (3).
http_proxy	string	Ciąg połączenia proxy HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
headers		Element główny nagłówków żądania HTTP(S). Obsługiwane dla pozycji HTTP_AGENT.
name	string	(wymagane dla pozycji HTTP_AGENT) Nazwa nagłówka.
value	string	(wymagane dla pozycji HTTP_AGENT) Wartość nagłówka.
retrieve_mode	string	Część odpowiedzi, która powinna zostać zapisana. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
request_method	string	Typ metody żądania. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] GET (0, domyślnie), POST (1), PUT (2), HEAD (3).

Element	Typ	Opis
output_format	string	Sposób przetwarzania odpowiedzi. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] RAW (0, domyślnie), JSON (1).
allow_traps	string	Zezwalanie na wypełnianie wartości w taki sam sposób jak dla pozycji trappera. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
ssl_key_password	string	Hasło do pliku klucza SSL. Obsługiwane dla pozycji HTTP_AGENT.
verify_peer	string	Czy należy sprawdzić, czy certyfikat hosta jest autentyczny. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Czy należy sprawdzić, czy nazwa hosta dla połączenia jest zgodna z nazwą w certyfikacie hosta. Obsługiwane dla pozycji HTTP_AGENT. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
time_shift	string	Jak daleko wstecz (przed bieżącym czasem) kończy się zakres czasu, którego dotyczy zapytanie pozycji. Obsługiwane dla pozycji TELEMETRY_QUERY.
lookback_limit	string	Jak daleko wstecz (przed bieżącym czasem) pozycja wyszukuje dane, gdy nie ma poprzedniego zapytania, od którego można kontynuować (np. podczas pierwszego uruchomienia pozycji), lub gdy poprzednie zapytanie jest starsze niż `delay` (np. z powodu niedostępności serwera Zabbix). Obsługiwane dla pozycji TELEMETRY_QUERY.
granularity	string	Długość fragmentu, na który dzielony jest pełny zakres czasu zapytania. Obsługiwane dla pozycji TELEMETRY_QUERY.
query		(wymagane dla pozycji TELEMETRY_QUERY) Element główny zapytania telemetrycznego. Określa, jakie dane pozycja pobiera ze źródła danych, jak je filtruje i jak agreguje wynik.
signal_type	string	(wymagane) Typ sygnału, którego dotyczy zapytanie pozycji. Obsługiwane dla pozycji TELEMETRY_QUERY. Możliwe wartości:[1] TRACES (0), METRICS (1), LOGS (2).
metric_point_type	string	(wymagane dla typu sygnału METRICS) Typ punktu metryki, którego dotyczy zapytanie pozycji. Obsługiwane dla pozycji TELEMETRY_QUERY. Możliwe wartości:[1] SUM (0), GAUGE (1), HISTOGRAM (2), EXPONENTIAL_HISTOGRAM (3).
columns		(wymagane) Element główny kolumn danych telemetrycznych, których dotyczy zapytanie pozycji. Ten element zawiera następujące elementy: `column (string)` - (wymagane) Nazwa kolumny danych telemetrycznych, której dotyczy zapytanie pozycji. `attribute_key (string)` - (wymagane dla kolumn zawierających pary klucz-wartość) Nazwa klucza, którego dotyczy zapytanie pozycji, w kolumnie zawierającej pary klucz-wartość.
aggregated_columns		(wymagane) Element główny zagregowanych kolumn danych telemetrycznych, których dotyczy zapytanie pozycji. Ten element zawiera następujące elementy: `column (string)` - (wymagane dla funkcji MIN, MAX, AVG, SUM i PERCENTILE) Nazwa kolumny danych telemetrycznych, której dotyczy zapytanie pozycji. `function (string)` - (wymagane) Funkcja stosowana przez pozycję do rekordów kolumny. Możliwe wartości:[1] MIN (1), MAX (2), AVG (3), COUNT (4), SUM (5), PERCENTILE (8). `parameters (array)` - (wymagane dla funkcji PERCENTILE) Percentyl obliczany przez pozycję na podstawie rekordów kolumny. `alias (string)` - (wymagane) Unikalna nazwa obliczonej wartości, używana do identyfikowania jej w wartości pozycji.

Element	Typ	Opis
filter		(wymagane) Element główny filtra zapytania telemetrycznego dla rekordów kolumn w zapytaniu. Ten element zawiera następujące elementy: `evaltype` (`string`) - (wymagane) Metoda oceny warunków filtra. Możliwe wartości:[1] AND_OR (0), AND (1), OR (2), FORMULA (3). `formula` (`string`) - (wymagane dla metod oceny FORMULA) Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów za pomocą niestandardowego wyrażenia. `conditions` (`string`) - (wymagane) Element główny warunków filtra danych telemetrycznych, które musi spełniać rekord w kolumnie danych telemetrycznych. Element `conditions` zawiera następujące elementy: `column` (`string`) - (wymagane) Nazwa kolumny danych telemetrycznych, której dotyczą warunki filtra. `attribute_key` (`string`) - (wymagane dla kolumn zawierających pary klucz-wartość) Nazwa klucza, którego dotyczy warunek, w kolumnie zawierającej pary klucz-wartość. `operator` (`string`) - (wymagane) Operator warunku filtra. Możliwe wartości dla kolumn zawierających pojedyncze wartości:[1] EQUAL (0), NOT_EQUAL (1), LIKE (2), NOT_LIKE (3). Możliwe wartości dla kolumn zawierających pary klucz-wartość:[1] EQUAL (0), NOT_EQUAL (1), EXISTS (12). `formulaid` (`string`) - (wymagane dla metod oceny FORMULA) Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. `value` (`string`) - (wymagane dla operacji EQUAL, NOT_EQUAL, LIKE i NOT LIKE) Wartość do porównania.
tags		Element główny tagów pozycji.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
triggers		Element główny wyzwalaczy pozycji hosta.

Note:
Zobacz także: Obiekt item (odpowiednią właściwość o pasującej nazwie).

Kroki przetwarzania wstępnego wartości pozycji hosta

```
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	Opis
type	string	(wymagane) Typ kroku przetwarzania wstępnego wartości pozycji. Dostępne wartości:[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, obliczane: otrzymana wartość - poprzednia wartość), CHANGE_PER_SECOND (10, obliczane: (otrzymana wartość - poprzednia wartość)/(czas bieżący - czas ostatniego sprawdzenia)), 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		(wymagane) Element główny dla parametrów kroku przetwarzania wstępnego wartości pozycji.
parameter	string	Pojedynczy parametr kroku przetwarzania wstępnego wartości pozycji.
error_handler	string	Typ akcji używany w przypadku niepowodzenia kroku przetwarzania wstępnego. Dostępne wartości:[1] ORIGINAL_ERROR (0, domyślnie), DISCARD_VALUE (1), CUSTOM_VALUE (2), CUSTOM_ERROR (3).
error_handler_params	string	Parametry obsługi błędów.

Note:
Zobacz też: Obiekt przetwarzania wstępnego pozycji (odwołanie do odpowiedniej właściwości o zgodnej nazwie).

Wyzwalacze pozycji hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      items:
        (...)
        - name: Uptime
          (...)
          triggers:
            - expression: 'nodata(/MySQL server/mysql.uptime,30m)=1'
              name: 'MySQL: Nie udało się pobrać danych informacyjnych'
              event_name: 'MySQL: Nie udało się pobrać danych informacyjnych (lub brak danych przez 30m)'
              priority: INFO
              description: 'Zabbix nie otrzymał żadnych danych dla pozycji w ciągu ostatnich 30 minut.'
              dependencies:
                - name: 'MySQL: Usługa jest niedostępna'
                  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: Usługa została uruchomiona ponownie'
              event_name: 'MySQL: Usługa została uruchomiona ponownie (uptime < 10m)'
              priority: INFO
              description: 'Czas działania MySQL jest krótszy niż 10 minut.'
              tags:
                - tag: scope
                  value: notice
        (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikatowy identyfikator tego wyzwalacza.
expression	string	(wymagane) Wyrażenie wyzwalacza.

Element	Type	Opis
recovery_mode	string	Podstawa generowania zdarzeń OK. Możliwe wartości:[1] EXPRESSION (0, domyślnie), RECOVERY_EXPRESSION (1), NONE (2).
recovery_expression	string	Wyrażenie odzyskania wyzwalacza.
correlation_mode	string	Tryb korelacji (bez korelacji zdarzeń lub korelacja zdarzeń według tagu). Możliwe wartości:[1] DISABLED (0, domyślnie), TAG_VALUE (1).
correlation_tag	string	Nazwa tagu używana do korelacji zdarzeń.
name	string	(wymagane) Nazwa wyzwalacza.
event_name	string	Nazwa zdarzenia.
opdata	string	Dane operacyjne.
url_name	string	Etykieta adresu URL powiązanego z wyzwalaczem.
url	string	Adres URL powiązany z wyzwalaczem.
status	string	Status wyzwalacza. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
priority	string	Ważność wyzwalacza. Możliwe wartości:[1] NOT_CLASSIFIED (0, domyślnie), INFO (1), WARNING (2), AVERAGE (3), HIGH (4), DISASTER (5).
description	text	Opis wyzwalacza.
type	string	Typ generowania zdarzeń (pojedyncze zdarzenie problemu lub wiele zdarzeń problemu). Możliwe wartości:[1] SINGLE (0, domyślnie), MULTIPLE (1).
manual_close	string	Ręczne zamykanie zdarzeń problemu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
dependencies		Element główny dla zależności.
name	string	(wymagane) Nazwa wyzwalacza zależnego.
expression	string	(wymagane) Wyrażenie wyzwalacza zależnego.
recovery_expression	string	Wyrażenie odzyskania wyzwalacza zależnego.
tags		Element główny dla tagów wyzwalacza.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.

Note:
Zobacz też: Trigger object (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Reguły wykrywania niskiego poziomu hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      discovery_rules:
        - name: 'Wykrywanie baz danych'
          key: 'mysql.db.discovery["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}"]'
          delay: 1h
          filter: (patrz tabela poniżej)
          description: 'Skanowanie baz danych w DBMS.'
          interface_ref: if1
          item_prototypes:
            - name: 'Rozmiar bazy danych {#DATABASE}'
              key: 'mysql.db.size["{$MYSQL.DSN}","{$MYSQL.USER}","{$MYSQL.PASSWORD}","{#DATABASE}"]'
              delay: 5m
              units: B
              description: 'Rozmiar bazy danych.'
              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:
Większość elementów reguły wykrywania niskiego poziomu hosta jest taka sama jak w przypadku pozycji hosta. Poniższa tabela opisuje te elementy, które różnią się od pozycji hosta.

Element	Type	Opis
type	string	Typ pozycji. Dopuszczalne wartości:[1] ZABBIX_PASSIVE (0, domyślnie), 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	(wymagane) Klucz reguły wykrywania niskiego poziomu.
filter		Element główny dla filtrów reguły wykrywania niskiego poziomu hosta.
lifetime	string	Okres czasu (w sekundach, z sufiksem czasu lub makrem użytkownika), po którym zasoby nieodnalezione już nie będą usuwane. Domyślnie: 7d.
lifetime_type	string	Scenariusz usuwania utraconych zasobów LLD. Dopuszczalne wartości: DELETE_NEVER, DELETE_IMMEDIATELY, DELETE_AFTER.
enabled_lifetime	string	Okres czasu (w sekundach, z sufiksem czasu lub makrem użytkownika), po którym nieodnalezione już zasoby zostaną wyłączone.
enabled_lifetime_type	string	Scenariusz wyłączania utraconych zasobów LLD. Dopuszczalne wartości: DISABLE_NEVER, DISABLE_IMMEDIATELY, DISABLE_AFTER.
item_prototypes		Element główny dla elementów prototypów pozycji hosta, które są takie same jak dla pozycji hosta.
trigger_prototypes		Element główny dla prototypów wyzwalaczy hosta, które są takie same jak dla wyzwalaczy pozycji hosta.
graph_prototypes		Element główny dla prototypów wykresów hosta, które są takie same jak dla wykresów hosta.
host_prototypes		Element główny dla prototypów hostów, które są takie same jak dla hostów.
parent_discovery_rule		Element główny dla nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły) prototypu reguły wykrywania niskiego poziomu. Ta właściwość oznacza, że jest to prototyp reguły LLD, bezpośredni element podrzędny wskazanej reguły/prototypu reguły.
key	string	(wymagane) Klucz nadrzędnej reguły wykrywania niskiego poziomu (lub prototypu reguły).
master_item	string	(wymagane dla reguł DEPENDENT) Element główny dla elementu nadrzędnego reguły zależnej.
lld_macro_paths		Element główny dla ścieżek makr reguły wykrywania niskiego poziomu.
lld_macro	string	(wymagane) Nazwa makra reguły wykrywania niskiego poziomu.
path	string	(wymagane) Selektor wartości, która zostanie przypisana do odpowiedniego makra.
preprocessing		Element główny dla wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu.
step		Element główny dla elementów kroku wstępnego przetwarzania wartości reguły wykrywania niskiego poziomu, które są takie same jak dla kroków wstępnego przetwarzania wartości pozycji hosta, z wyjątkiem mniejszej liczby możliwych wartości. Zobacz też: obiekt wstępnego przetwarzania reguły LLD.
overrides		Element główny dla reguł nadpisywania reguły wykrywania niskiego poziomu.
name	string	(wymagane) Unikalna nazwa nadpisania.
step	string	(wymagane) Unikalny numer kolejności nadpisania.
stop	string	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli pasuje.
filter		Element główny dla elementów filtra reguły nadpisywania reguły wykrywania niskiego poziomu, które są takie same jak dla filtrów reguły wykrywania niskiego poziomu hosta.

Element	Type	Opis
operations		Element główny dla operacji nadpisywania reguły wykrywania niskiego poziomu hosta.

Note:
Zobacz też: obiekt reguły LLD (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Filtry reguły wykrywania niskiego poziomu hosta

```
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	Opis
evaltype	string	Zastępuje metodę oceny warunków filtra. Dopuszczalne wartości:[1] AND_OR (0, domyślnie), AND (1), OR (2), FORMULA (3).
formula	string	Niestandardowa formuła obliczeniowa dla warunków filtra.
conditions		Element nadrzędny dla warunków filtra.
macro	string	(wymagane) Nazwa makra wykrywania niskiego poziomu, na którym ma zostać wykonane sprawdzenie.
value	string	Wartość do porównania.
operator	string	Operator warunku. Dopuszczalne wartości:[1] MATCHES_REGEX (8, domyślnie), NOT_MATCHES_REGEX (9).
formulaid	string	(wymagane) Dowolny unikalny identyfikator używany do odwołania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale przy kolejnym pobraniu zostanie wygenerowany ponownie.

Note:
Zobacz też: obiekt filtra reguły LLD (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Operacje nadpisywania reguły wykrywania niskiego poziomu hosta

Element	Type	Description
operationobject	string	Obiekt, do którego jest stosowana operacja. Możliwe wartości:[1] ITEM_PROTOTYPE (0), TRIGGER_PROTOTYPE (1), GRAPH_PROTOTYPE (2), HOST_PROTOTYPE (3), DISCOVERY_PROTOTYPE (4).
operator	string	Operator warunku nadpisania. Możliwe wartości:[1] EQUAL (1), NOT_EQUAL (2), LIKE (3), NOT_LIKE (4), REGEXP (5), NOT_REGEXP (6).
value	string	Wyrażenie regularne lub ciąg znaków dla operatora warunku nadpisania.
status	string	Status obiektu po wykonaniu operacji nadpisania.
discover	string	Czy obiekt jest dodawany w wyniku wykrycia.
delay	string	Interwał aktualizacji ustawiony dla prototypu pozycji po wykonaniu operacji nadpisania.
history	string	Okres przechowywania historii ustawiony dla prototypu pozycji po wykonaniu operacji nadpisania.

Element	Type	Description
trends	string	Okres przechowywania trendów ustawiony dla prototypu pozycji po wykonaniu operacji nadpisania.
severity	string	Poziom ważności prototypu wyzwalacza ustawiony po wykonaniu operacji nadpisania.
tags		Element główny dla tagów ustawionych dla obiektu po wykonaniu operacji nadpisania.
tag	string	(wymagane) Nazwa tagu.
value	string	Wartość tagu.
templates		Element główny dla szablonów powiązanych z prototypem hosta po wykonaniu operacji nadpisania.
name	string	(wymagane) Nazwa szablonu.
inventory_mode	string	Tryb inwentarza prototypu hosta ustawiony po wykonaniu operacji nadpisania.

Note:
Zobacz także: Obiekt operacji nadpisywania reguły LLD (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Scenariusze web hosta

Element	Type	Description
uuid	string	(required) Unikalny identyfikator tego scenariusza web.
name	string	(required) Nazwa scenariusza web.
delay	string	Częstotliwość (w sekundach, sufiks czasu lub makro użytkownika) wykonywania scenariusza web. Domyślnie: 1m.
attempts	integer	Liczba prób wykonania kroków scenariusza web. Możliwe wartości: 1-10 (domyślnie: 1).
agent	string	Agent klienta. Zabbix będzie udawał wybraną przeglądarkę. Jest to przydatne, gdy witryna zwraca różną zawartość dla różnych przeglądarek. Domyślnie: Zabbix.
http_proxy	string	Proxy używany przez scenariusz web, podany jako: `http://[username[:password]@]proxy.example.com[:port]`
variables		Element główny dla zmiennych scenariusza web, które mogą być używane w krokach scenariusza.
name	string	(required) Nazwa zmiennej.
value	text	(required) Wartość zmiennej.
headers		Element główny dla nagłówków HTTP wysyłanych podczas wykonywania żądania. Nagłówki powinny być wymienione przy użyciu tej samej składni, w jakiej występowałyby w protokole HTTP.
name	string	(required) Nazwa nagłówka.
value	text	(required) Wartość nagłówka.
status	string	Status scenariusza web. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1).
authentication	string	Metoda uwierzytelniania. Możliwe wartości:[1] NONE (0, domyślnie), BASIC (1), NTLM (2), Kerberos (3) lub Digest (4).
http_user	string	Nazwa użytkownika używana dla uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
http_password	string	Hasło używane dla uwierzytelniania BASIC (HTTP), NTLM, Kerberos lub Digest.
verify_peer	string	Weryfikuj certyfikat SSL serwera web. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
verify_host	string	Weryfikuj, czy pole *Common Name* lub pole *Subject Alternate Name* certyfikatu serwera web jest zgodne. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
ssl_cert_file	string	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_file	string	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_password	string	Hasło pliku prywatnego klucza SSL.
steps		(required) Element główny dla kroków scenariusza web hosta.
tags		Element główny dla tagów scenariusza web.
tag	string	(required) Nazwa tagu.

Element	Type	Description
value	string	Wartość tagu.

Note:
Zobacz także: Obiekt scenariusza web (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Kroki scenariusza web hosta

Element	Type	Description
name	string	(wymagane) Nazwa kroku scenariusza web.
url	string	(wymagane) URL do monitorowania.
query_fields		Element główny dla parametrów zapytania (tablica pól HTTP dodawanych do URL podczas wykonywania żądania).
name	string	(wymagane) Nazwa parametru zapytania.
value	string	Wartość parametru zapytania.
posts		Element główny dla zmiennych HTTP POST (ciąg znaków (surowe dane post) lub tablica pól HTTP (dane pól formularza)).
name	string	(wymagane) Nazwa pola post.
value	string	(wymagane) Wartość pola post.
variables		Element główny zmiennych na poziomie kroku (makr), które mają zostać zastosowane po tym kroku. Jeśli wartość zmiennej ma prefiks 'regex:', jej wartość jest wyodrębniana z danych zwróconych przez ten krok zgodnie z wyrażeniem regularnym następującym po prefiksie 'regex:'
name	string	(wymagane) Nazwa zmiennej.
value	text	(wymagane) Wartość zmiennej.
headers		Element główny nagłówków HTTP, które mają zostać wysłane podczas wykonywania żądania.
name	string	(wymagane) Nazwa nagłówka.
value	text	(wymagane) Wartość nagłówka.
follow_redirects	string	Podążanie za przekierowaniami HTTP. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
retrieve_mode	string	Tryb pobierania odpowiedzi HTTP. Możliwe wartości:[1] BODY (0, domyślnie), HEADERS (1), BOTH (2).
timeout	string	Limit czasu (w sekundach, przyrostek czasu lub makro użytkownika) wykonania kroku. Domyślnie: 15s.
required	string	Tekst, który musi być obecny w odpowiedzi (ignorowany, jeśli pusty).
status_codes	string	Lista kodów statusu HTTP rozdzielona przecinkami, które są akceptowane (np. 200-201,210-299; ignorowane, jeśli puste).

Note:
Zobacz także: Obiekt kroku scenariusza web (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Wykresy hosta

```
zabbix_export:
  (...)
  graphs:
    - name: 'MySQL: Bandwidth'
      graph_items: (see table below)
    (...)
```

Element	Type	Opis
uuid	string	Unikalny identyfikator wykresu.
name	string	(wymagane) Nazwa wykresu.
width	integer	Szerokość wykresu w pikselach. Używane do podglądu oraz dla wykresów kołowych/rozszerzonych. Możliwe wartości:[1] 20-65535 (domyślnie: 900).

Element	Type	Opis
height	integer	Wysokość wykresu w pikselach. Używane do podglądu oraz dla wykresów kołowych/rozszerzonych. Możliwe wartości:[1] 20-65535 (domyślnie: 900).
yaxismin	double	Wartość minimum osi Y. Obsługiwane dla stałej wartości minimalnej osi Y. Domyślnie: 0.
yaxismax	double	Wartość maksimum osi Y. Obsługiwane dla stałej wartości minimalnej osi X. Domyślnie: 0.
show_work_period	string	Wyróżnij godziny niepracujące. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
show_triggers	string	Wyświetlaj proste wartości wyzwalaczy jako linię. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
type	string	Typ wykresu. Możliwe wartości:[1] NORMAL (0, domyślnie), STACKED (1), PIE (2), EXPLODED (3).
show_legend	string	Wyświetl legendę wykresu. Możliwe wartości:[1] NO (0), YES (1, domyślnie).
show_3d	string	Włącz styl 3D. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
percent_left	double	Pokaż linię percentyla dla lewej osi. Obsługiwane dla wykresów NORMAL. Domyślnie: 0.
percent_right	double	Pokaż linię percentyla dla prawej osi. Obsługiwane dla wykresów NORMAL. Domyślnie: 0.
ymin_type_1	string	Minimalna wartość osi Y. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] CALCULATED (0, domyślnie), FIXED (1), ITEM (2).
ymin_item_1		(wymagane, jeśli `ymin_type_1` jest ustawione na `ITEM`) Element główny dla szczegółów pojedynczej pozycji.
host	string	(wymagane) host pozycji.
key	string	(wymagane) klucz pozycji.
ymax_type_1	string	Maksymalna wartość osi Y. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] CALCULATED (0, domyślnie), FIXED (1), ITEM (2).
ymax_item_1		(wymagane, jeśli `ymax_type_1` jest ustawione na `ITEM`) Element główny dla szczegółów pojedynczej pozycji.
host	string	(wymagane) host pozycji.
key	string	(wymagane) klucz pozycji.
graph_items		(wymagane) Element główny dla elementów wykresu hosta.

Note:
Zobacz też: Obiekt Graph (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Elementy wykresu hosta

```
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
    (...)
```

Element	Type	Opis
sortorder	integer	Kolejność rysowania. Mniejsza wartość jest rysowana jako pierwsza. Może być używana do rysowania linii lub obszarów za (lub przed) innymi.
drawtype	string	Styl rysowania elementu wykresu. Obsługiwane dla wykresów NORMAL. Możliwe wartości:[1] SINGLE_LINE (0, domyślnie), FILLED_REGION (1), BOLD_LINE (2), DOTTED_LINE (3), DASHED_LINE (4), GRADIENT_LINE (5).
color	string	Kolor elementu (6 znaków, hex).
yaxisside	string	Strona wykresu, po której będzie rysowana skala Y elementu wykresu. Obsługiwane dla wykresów NORMAL i STACKED. Możliwe wartości:[1] LEFT (0, domyślnie), RIGHT (1).
calc_fnc	string	Dane do wyświetlenia, jeśli dla pozycji istnieje więcej niż jedna wartość. Możliwe wartości:[1] MIN (1), AVG (2, domyślnie), MAX (4), ALL (7; minimum, średnia i maksimum; obsługiwane dla prostych wykresów), LAST (9, obsługiwane dla wykresów kołowych/rozszerzonych).
type	string	Typ elementu wykresu. Możliwe wartości:[1] SIMPLE (0, domyślnie), GRAPH_SUM (2; wartość pozycji reprezentuje cały wykres kołowy; obsługiwane dla wykresów kołowych/rozszerzonych).
item		(wymagane) Pojedyncza pozycja.
host	string	(wymagane) Host pozycji.
key	string	(wymagane) Klucz pozycji.

Note:
Zobacz też: Obiekt elementu wykresu (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Mapy wartości hosta

```
zabbix_export:
  (...)
  hosts:
    - host: 'MySQL server'
      (...)
      valuemaps:
        - name: Example value map
          mappings:
            - value: '1'
              newvalue: Example value
  (...)
```

Element	Type	Opis
uuid	string	(wymagane) Unikalny identyfikator tej mapy wartości.
name	string	(wymagane) Nazwa mapy wartości.
mapping		Element główny dla mapowań.
type	string	Typ dopasowania mapowania. Dopuszczalne wartości:[1] EQUAL (0, domyślnie), GREATER_OR_EQUAL (2), LESS_OR_EQUAL (3), IN_RANGE (4), REGEXP (5), DEFAULT (6).
value	string	Wartość oryginalna.
newvalue	string	(wymagane) Wartość, na którą mapowana jest wartość oryginalna.

Note:
Zobacz także: Obiekt mapy wartości (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Przypisy [1] Wartości całkowite API w nawiasach, na przykład ENABLED (0), są podane wyłącznie jako informacja pomocnicza. Więcej informacji można znaleźć na powiązanej stronie obiektu API w wpisie tabeli lub na końcu każdej sekcji.

Mapy sieciowe

Przegląd

Mapa sieci eksport zawiera:

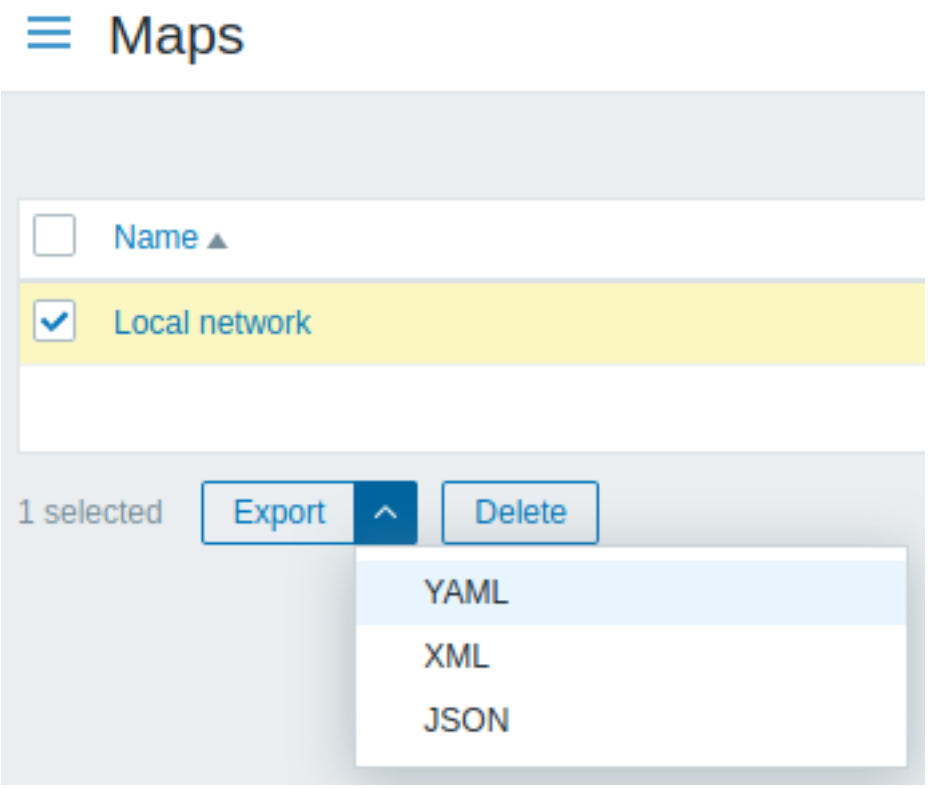
- Wszystkie powiązane obrazy
- Strukturę mapy (wszystkie ustawienia mapy, wszystkie zawarte elementy wraz z ich ustawieniami, połączenia mapy oraz wskaźniki stanu połączeń mapy)

Warning:
Żadne grupy hostów, hosty, wyzwalacze, inne mapy ani inne elementy, które mogą być powiązane z eksportowaną mapą, nie są eksportowane. W związku z tym, jeśli brakuje co najmniej jednego z elementów, do których odwołuje się mapa, jej import zakończy się niepowodzeniem.

Eksportowanie

Aby wyeksportować mapy sieci, wykonaj następujące czynności:

1. Przejdź do *Monitoring* > *Maps*.
2. Zaznacz pola wyboru map sieci, które chcesz wyeksportować.
3. Kliknij *Export* pod listą, aby wyeksportować w domyślnym formacie YAML, lub kliknij strzałkę, aby wybrać inny format eksportu.



W zależności od wybranego formatu mapy są eksportowane do pliku lokalnego o domyślnej nazwie:

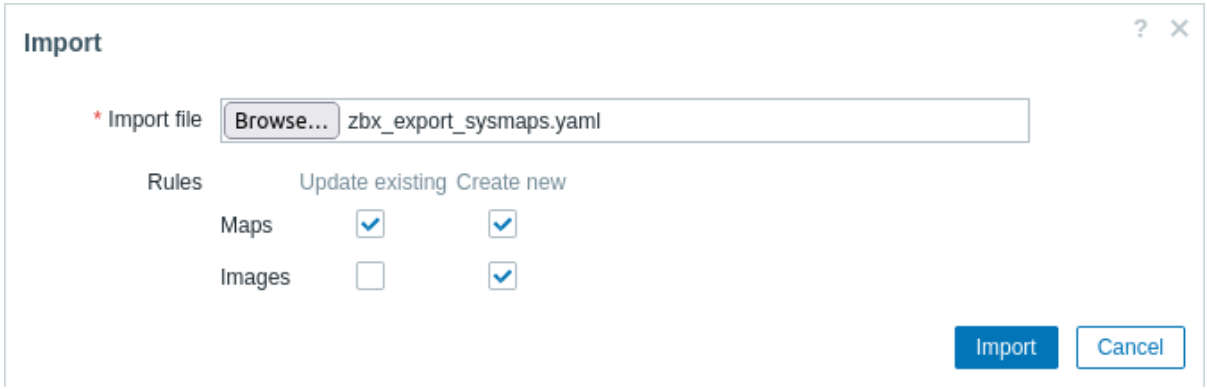
- `zabbix_export_maps.yaml` - w eksporcie YAML;
- `zabbix_export_maps.xml` - w eksporcie XML;
- `zabbix_export_maps.json` - w eksporcie JSON.

Importowanie

Aby zaimportować mapy sieciowe, wykonaj następujące czynności:

1. Przejdź do *Monitoring* > *Maps*.
2. Kliknij *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz pola wyboru reguł importu, które chcesz zastosować (zobacz poniżej).

5. Kliknij *Import* w prawym dolnym rogu formularza konfiguracji.



Reguły importu:

Reguła	Opis
Update existing	Istniejące mapy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe mapy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.

Jeśli odznaczysz obie opcje dotyczące map i zaznaczysz odpowiednie opcje dotyczące obrazów, zostaną zaimportowane tylko obrazy. Importowanie obrazów jest dostępne tylko dla użytkowników z rolą *Super admin*.

Podczas importu nie można przejrzeć zmian w mapach sieciowych. Komunikat o pomyślnym lub nieudanym imporcie zostanie wyświetlony w frontend.

Warning:
Zastąpienie istniejącego obrazu wpłynie na wszystkie mapy, które używają tego obrazu.

Format eksportu

Eksport do YAML:

```yaml
zabbix_export:
  version: '8.0'
  images:
    - name: Zabbix_server_3D_(128)
      imagetype: '1'
      encodedImage: iVBOR...5CYII=
  maps:
    - name: 'Local network'
      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: {  }
```

Eksportowane elementy

Eksportowane elementy są opisane w poniższej tabeli.

Element	Type	Description
images		Element główny dla obrazów.
name	string	Unikalna nazwa obrazu.
imagetype	integer	Typ obrazu. Dostępne wartości: 1 - Obraz; 2 - Tło.

Element	Type	Description
encodedImage	string	Obraz zakodowany w formacie Base64.
maps		Element główny dla map.

Mapy

Element	Type	Opis
name	string	Unikalna nazwa mapy.
width	integer	Szerokość mapy w pikselach.
height	integer	Wysokość mapy w pikselach.
label_type	integer	Typ etykiety elementu mapy. Dostępne wartości: 0 - Etykieta; 1 - Adres IP hosta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Brak.
label_location	integer	Domyślne położenie etykiety elementu mapy. Dostępne wartości: 0 - Dół; 1 - Lewo; 2 - Prawo; 3 - Góra.
highlight	integer	Włącz podświetlanie ikon dla aktywnych wyzwalaczy i statusów hostów. Dostępne wartości: 0 - Nie; 1 - Tak.
expandproblem	integer	Wyświetl wyzwalacz problemu dla elementów z pojedynczym problemem. Dostępne wartości: 0 - Nie; 1 - Tak.
markelements	integer	Podświetl elementy mapy, które niedawno zmieniły status. Dostępne wartości: 0 - Nie; 1 - Tak.
show_unack	integer	Wyświetlanie problemów. Dostępne wartości: 0 - Liczba wszystkich problemów; 1 - Liczba niepotwierdzonych problemów; 2 - Osobno liczba potwierdzonych i niepotwierdzonych problemów.
severity_min	integer	Minimalny poziom ważności wyzwalacza wyświetlany domyślnie na mapie. Dostępne wartości: 0 - Nie sklasyfikowano; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa.
show_suppressed	integer	Wyświetl problemy, które w przeciwnym razie byłyby ukryte (niewyświetlane) z powodu konserwacji hosta. Dostępne wartości: 0 - Nie; 1 - Tak.
grid_size	integer	Rozmiar komórki siatki mapy w pikselach. Obsługiwane, jeśli `grid_show` ma wartość 0. Dostępne wartości: 20, 40, 50, 75 lub 100.
grid_show	integer	Wyświetl siatkę w konfiguracji mapy. Dostępne wartości: 0 - Tak; 1 - Nie.

Element	Type	Opis
grid_align	integer	Automatycznie wyrównaj ikony w konfiguracji mapy. Dostępne wartości: 0 - Tak; 1 - Nie.
label_format	integer	Użyj zaawansowanej konfiguracji etykiet. Dostępne wartości: 0 - Nie; 1 - Tak.
label_type_host	integer	Wyświetl etykietę jako etykietę hosta. Obsługiwane, jeśli `label_format` ma wartość 1. Dostępne wartości: 0 - Etykieta; 1 - Adres IP hosta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Brak; 5 - Etykieta niestandardowa.
label_type_hostgroup	integer	Wyświetl etykietę jako etykietę grupy hostów. Obsługiwane, jeśli `label_format` ma wartość 1. Dostępne wartości: 0 - Etykieta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Brak; 5 - Etykieta niestandardowa.
label_type_trigger	integer	Wyświetl etykietę jako etykietę wyzwalacza. Obsługiwane, jeśli `label_format` ma wartość 1. Dostępne wartości: 0 - Etykieta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Brak; 5 - Etykieta niestandardowa.
label_type_map	integer	Wyświetl etykietę jako etykietę mapy. Obsługiwane, jeśli `label_format` ma wartość 1. Dostępne wartości: 0 - Etykieta; 2 - Nazwa elementu; 3 - Tylko status; 4 - Brak; 5 - Etykieta niestandardowa.
label_type_image	integer	Wyświetl etykietę jako etykietę obrazu. Obsługiwane, jeśli `label_format` ma wartość 1. Dostępne wartości: 0 - Etykieta; 2 - Nazwa elementu; 4 - Brak; 5 - Etykieta niestandardowa.
label_string_host	string	Niestandardowa etykieta dla elementów hosta. Obsługiwane, jeśli `label_type_host` ma wartość 5.
label_string_hostgroup	string	Niestandardowa etykieta dla elementów grupy hostów. Obsługiwane, jeśli `label_type_hostgroup` ma wartość 5.
label_string_trigger	string	Niestandardowa etykieta dla elementów wyzwalacza. Obsługiwane, jeśli `label_type_trigger` ma wartość 5.
label_string_map	string	Niestandardowa etykieta dla elementów mapy. Obsługiwane, jeśli `label_type_map` ma wartość 5.
label_string_image	string	Niestandardowa etykieta dla elementów obrazu. Obsługiwane, jeśli `label_type_image` ma wartość 5.
expand_macros	integer	Rozwijaj makra w etykietach w konfiguracji mapy. Dostępne wartości: 0 - Nie; 1 - Tak.

Element	Type	Opis
background		Element główny dla obrazu tła (jeśli istnieje). Obsługiwane, jeśli `imagetype` ma wartość 2.
name	string	Nazwa obrazu tła.
iconmap		Element główny dla mapowania ikon (jeśli istnieje).
name	string	Nazwa mapowania ikon.
urls		Element główny dla adresów URL używanych przez mapy lub poszczególne elementy mapy.
name	string	Nazwa łącza.
url	string	Adres URL łącza.
elementtype	integer	Typ elementu mapy, do którego należy łącze. Dostępne wartości: 0 - Host; 1 - Mapa; 2 - Wyzwalacz; 3 - Grupa hostów; 4 - Obraz.
selements		Element główny dla selementów mapy.
shapes		Element główny dla kształtów mapy.
type	integer	Typ kształtu. Dostępne wartości: 0 - Prostokąt; 1 - Elipsa.
x	integer	Współrzędna X kształtu w pikselach.
y	integer	Współrzędna Y kształtu w pikselach.
width	integer	Szerokość kształtu.
height	integer	Wysokość kształtu.
text	string	Tekst wewnątrz kształtu.
font	integer	Styl czcionki tekstu. Dostępne wartości: 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	Rozmiar czcionki w pikselach.
font_color	string	Kolor czcionki zapisany w kodzie szesnastkowym.
text_halign	integer	Poziome wyrównanie tekstu. Dostępne wartości: 0 - Środek; 1 - Lewo; 2 - Prawo.
text_valign	integer	Pionowe wyrównanie tekstu. Dostępne wartości: 0 - Środek; 1 - Góra; 2 - Dół.
border_type	integer	Typ obramowania kształtu. Dostępne wartości: 0 - Brak; 1 - Linia pogrubiona; 2 - Linia kropkowana; 3 - Linia kreskowana.
border_width	integer	Szerokość obramowania w pikselach.

Element		Type	Opis
	border_color	string	Kolor obramowania zapisany w kodzie szesnastkowym.
	background_color	string	Kolor tła (wypełnienia) zapisany w kodzie szesnastkowym.
	zindex	integer	Wartość określająca kolejność wszystkich kształtów i linii (z-index).
lines			Element główny dla linii mapy.
	x1	integer	Współrzędna X punktu 1 linii w pikselach.
	y1	integer	Współrzędna Y punktu 1 linii w pikselach.
	x2	integer	Współrzędna X punktu 2 linii w pikselach.
	y2	integer	Współrzędna Y punktu 2 linii w pikselach.
	line_type	integer	Typ linii. Dostępne wartości: 0 - Brak; 1 - Linia pogrubiona; 2 - Linia kropkowana; 3 - Linia kreskowana.
	line_width	integer	Szerokość linii w pikselach.
	line_color	string	Kolor linii zapisany w kodzie szesnastkowym.
	zindex	integer	Wartość określająca kolejność wszystkich kształtów i linii (z-index).
links			Element główny dla łączy między elementami mapy.
	drawtype	integer	Styl łącza. Dostępne wartości: 0 - Linia; 2 - Linia pogrubiona; 3 - Linia kropkowana; 4 - Linia kreskowana.
	color	string	Kolor łącza (6 znaków, hex).
	label	string	Etykieta łącza.
	selementid1	id	ID jednego elementu do połączenia.
	selementid2	id	ID drugiego elementu do połączenia.
linktriggers			Element główny dla wskaźników statusu łączy.

Note:
Zobacz też: Obiekt mapy (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Elementy mapy

Element	Type	Description
elementtype	integer	Typ elementu mapy. Możliwe wartości: 0 - Host; 1 - Mapa; 2 - Wyzwalacz; 3 - Grupa hostów; 4 - Obraz.
elements		Element główny dla encji Zabbix (host, grupa hostów, mapa itp.), które są reprezentowane na mapie. Każda encja ma odpowiadający jej element (`host` itd.; zobacz na przykład Format eksportu).
label	string	Etykieta ikony.
label_location	integer	Położenie etykiety. Możliwe wartości: -1 - Użyj domyślnego ustawienia mapy; 0 - Dół; 1 - Lewo; 2 - Prawo; 3 - Góra.
x	integer	Położenie na osi X.
y	integer	Położenie na osi Y.

Element	Type	Description
elementsubtype	integer	Podtyp elementu. Obsługiwane, jeśli `elementtype` ma wartość 3. Możliwe wartości: 0 - Pojedyncza grupa hostów; 1 - Wszystkie grupy hostów.
areatype	integer	Rozmiar obszaru. Obsługiwane, jeśli `elementtype` ma wartość 1. Możliwe wartości: 0 - Taki sam jak cała mapa; 1 - Rozmiar niestandardowy.
width	integer	Szerokość obszaru. Obsługiwane, jeśli `areatype` ma wartość 1.
height	integer	Wysokość obszaru. Obsługiwane, jeśli `areatype` ma wartość 1.
viewtype	integer	Algorytm rozmieszczenia obszaru. Obsługiwane, jeśli `elementsubtype` ma wartość 1. Możliwe wartości: 0 - Rozmieść równomiernie w obszarze.
use_iconmap	integer	Użyj mapowania ikon dla tego elementu. Istotne tylko wtedy, gdy mapowanie ikon jest aktywowane na poziomie mapy. Możliwe wartości: 0 - Nie; 1 - Tak.
selementid	id	Unikalny identyfikator rekordu elementu.
icon_off		Element główny dla obrazu używanego, gdy element ma stan `OK`.
name	string	Unikalna nazwa obrazu.
icon_on		Element główny dla obrazu używanego, gdy element ma stan `Problem`.
name	string	Unikalna nazwa obrazu.
icon_disabled		Element główny dla obrazu używanego, gdy element jest wyłączony.
name	string	Unikalna nazwa obrazu.
icon_maintenance		Element główny dla obrazu używanego, gdy element jest w trybie konserwacji.
name	string	Unikalna nazwa obrazu.
urls		Element główny dla adresów URL używanych przez mapy lub każdy element mapy.
name	string	Nazwa łącza.
url	string	Adres URL łącza.
evaltype	integer	Typ oceny dla tagów.
tags		Element główny dla tagów problemów (dla elementów host i grup hostów). Jeśli podano tagi, na mapie będą wyświetlane tylko problemy z tymi tagami.
tag	string	Nazwa tagu.
value	string	Wartość tagu.
operator	integer	Operator.

Note:
Zobacz także: Obiekt elementu mapy (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Wskaźniki stanu łącza mapy

Element	Type	Description
drawtype	integer	Styl łącza, gdy wyzwalacz znajduje się w stanie „Problem”. Dostępne wartości: 0 - Linia; 2 - Pogrubiona linia; 3 - Linia kropkowana; 4 - Linia przerywana.
color	string	Kolor łącza (6 znaków, hex), gdy wyzwalacz znajduje się w stanie „Problem”.
trigger		Element nadrzędny dla wyzwalacza używanego do wskazywania stanu łącza.
description	string	Nazwa wyzwalacza.
expression	string	Wyrażenie wyzwalacza.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza.

Note:
Zobacz także: Obiekt wyzwalacza łącza mapy (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Typy mediów

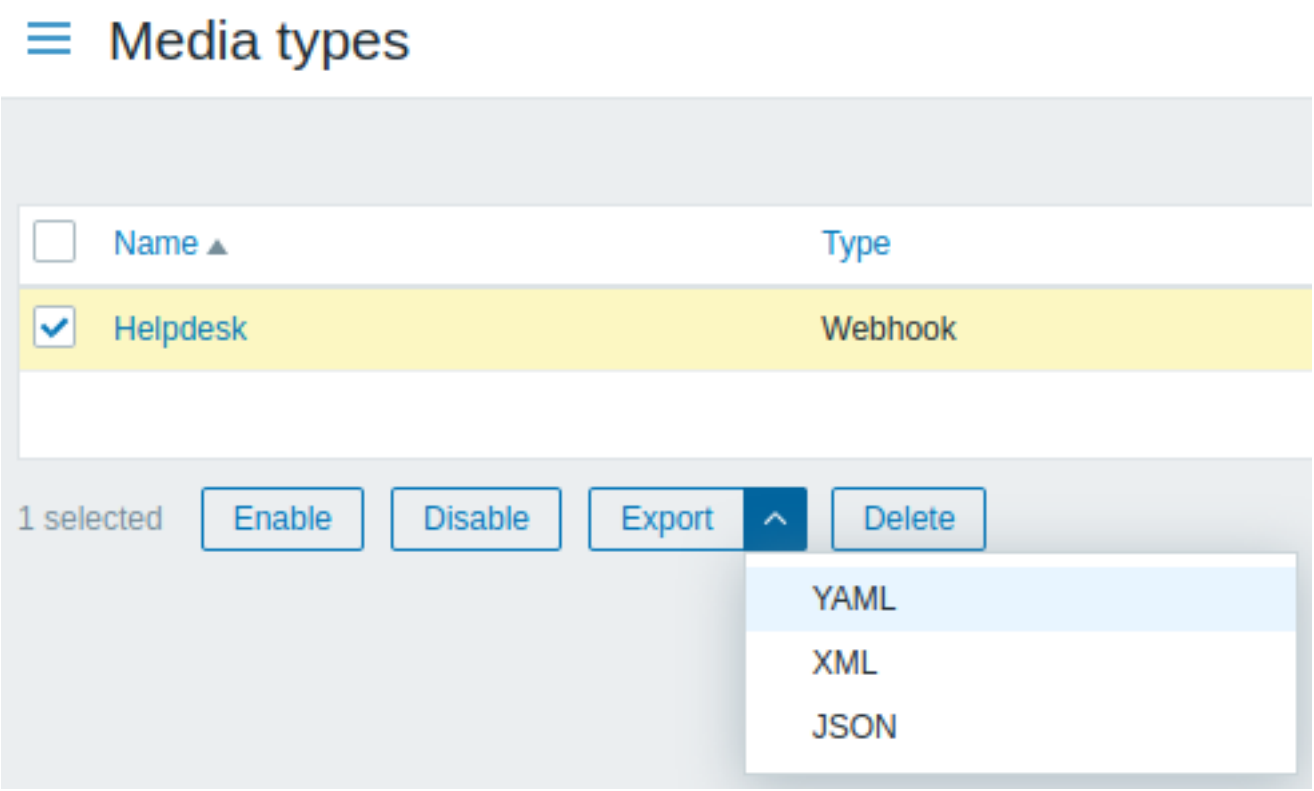
Przegląd

Typy mediów są eksportowane wraz ze wszystkimi powiązanymi obiektami i relacjami między obiektami.

Eksportowanie

Aby wyeksportować typy mediów, wykonaj następujące czynności:

1. Przejdź do *Alerts* > *Media types*.
2. Zaznacz pola wyboru przy typach mediów, które chcesz wyeksportować.
3. Kliknij *Export* pod listą, aby wyeksportować w domyślnym formacie YAML, lub kliknij strzałkę, aby wybrać inny format eksportu.



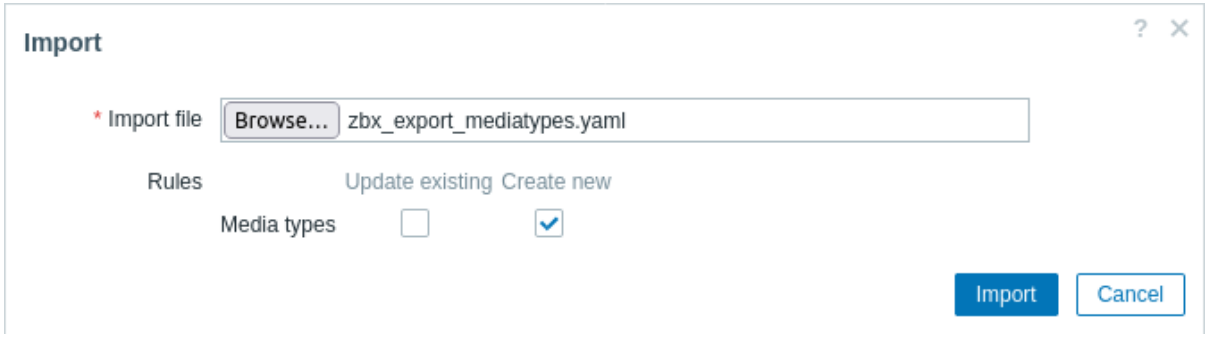
W zależności od wybranego formatu typy mediów są eksportowane do lokalnego pliku o domyślnej nazwie:

- `zabbix_export_mediatypes.yaml` - w eksporcie YAML;
- `zabbix_export_mediatypes.xml` - w eksporcie XML;
- `zabbix_export_mediatypes.json` - w eksporcie JSON.

Importowanie

Aby zaimportować typy mediów, wykonaj następujące czynności:

1. Przejdź do sekcji *Alerts* > *Media types*.
2. Kliknij przycisk *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz pola wyboru reguł importu, które chcesz zastosować (zobacz poniżej).
5. Kliknij przycisk *Import* w prawym dolnym rogu formularza konfiguracji.



Reguły importu:

Rule	Description
Update existing	Zaznacz to pole wyboru, aby zaktualizować istniejące typy mediów przy użyciu danych z pliku importu.
Create new	Zaznacz to pole wyboru, aby utworzyć nowe typy mediów przy użyciu danych z pliku importu. Jeśli plik importu zawiera typ mediów, który już istnieje (np. *Email*), nowy typ nie zostanie utworzony, a istniejący typ mediów również nie zostanie zaktualizowany.

Podczas importu nie można przejrzeć zmian dotyczących typów mediów. Komunikat o pomyślnym lub nieudanym imporcie zostanie wyświetlony w frontendzie.

Format eksportu

Eksport do 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 'Nieprawidłowy parametr "event_source": "' + params.event_source + '".\nMusi mieć wa
        }

        if (params.event_value !== '0' && params.event_value !== '1'
            && (params.event_source === '0' || params.event_source === '3')) {
            throw 'Nieprawidłowy parametr "event_value": ' + params.event_value + '\nMusi mieć wartość
        }

        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" powinno mieć wartość -2..2';
        }

        if (params.event_source === '0' && isNaN(params.triggerid)) {
            throw 'pole "triggerid" nie jest liczbą';
        }
```

```
if (isNaN(params.eventid)) {
    throw 'pole "eventid" nie jest liczbą';
}

if (typeof params.message !== 'string' || params.message.trim() === '') {
    throw 'pole "message" nie może być puste';
}

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 'pole "retry" powinno być liczbą o wartości co najmniej 30, jeśli "priority" jes
    }

    if (isNaN(params.expire) || params.expire > 10800) {
        throw 'pole "expire" powinno być liczbą o wartości co najwyżej 10800, jeśli "priority"
    }

    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: |
    Zapoznaj się z instrukcją konfiguracji tutaj: https://git.zabbix.com/projects/ZBX/repos/zabbix/bro

    Ustaw parametr token na klucz aplikacji Pushover.
    Przypisując media Pushover do użytkownika Zabbix - dodaj klucz użytkownika do pola Wyślij do.
message_templates:
  - event_source: TRIGGERS
    operation_mode: PROBLEM
    subject: 'Problem: {EVENT.NAME}'
    message: |
      Problem rozpoczął się o {EVENT.TIME} dnia {EVENT.DATE}
      Nazwa problemu: {EVENT.NAME}
      host: {HOST.NAME}
      Ważność: {EVENT.SEVERITY}
      Dane operacyjne: {EVENT.OPDATA}
      ID oryginalnego problemu: {EVENT.ID}
      {TRIGGER.URL}
  - event_source: TRIGGERS
    operation_mode: RECOVERY
    subject: 'Rozwiązano w {EVENT.DURATION}: {EVENT.NAME}'
    message: |
      Problem został rozwiązany o {EVENT.RECOVERY.TIME} dnia {EVENT.RECOVERY.DATE}
      Nazwa problemu: {EVENT.NAME}
      Czas trwania problemu: {EVENT.DURATION}
      host: {HOST.NAME}
      Ważność: {EVENT.SEVERITY}
      ID oryginalnego problemu: {EVENT.ID}
      {TRIGGER.URL}
  - event_source: TRIGGERS
    operation_mode: UPDATE
    subject: 'Zaktualizowany problem w {EVENT.AGE}: {EVENT.NAME}'
    message: |
      {USER.FULLNAME} {EVENT.UPDATE.ACTION} problem o {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}.
      {EVENT.UPDATE.MESSAGE}

      Bieżący stan problemu to {EVENT.STATUS}, wiek: {EVENT.AGE}, potwierdzony: {EVENT.ACK.STATUS}.
  - event_source: DISCOVERY
    operation_mode: PROBLEM
    subject: 'Odkrywanie: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}'
    message: |
      Reguła odkrywania: {DISCOVERY.RULE.NAME}

      Adres IP urządzenia: {DISCOVERY.DEVICE.IPADDRESS}
      DNS urządzenia: {DISCOVERY.DEVICE.DNS}
      Stan urządzenia: {DISCOVERY.DEVICE.STATUS}
      Czas działania urządzenia: {DISCOVERY.DEVICE.UPTIME}

      Nazwa usługi urządzenia: {DISCOVERY.SERVICE.NAME}
      Port usługi urządzenia: {DISCOVERY.SERVICE.PORT}
      Stan usługi urządzenia: {DISCOVERY.SERVICE.STATUS}
      Czas działania usługi urządzenia: {DISCOVERY.SERVICE.UPTIME}
  - event_source: AUTOREGISTRATION
    operation_mode: PROBLEM
    subject: 'Automatyczna rejestracja: {HOST.HOST}'
    message: |
      Nazwa hosta: {HOST.HOST}
      IP hosta: {HOST.IP}
```

```
            Port agenta: {HOST.PORT}
```

Wyeksportowane elementy

Wyeksportowane elementy są opisane w poniższej tabeli.

Element	Type	Opis
name	string	(wymagane) Nazwa typu mediów.
type	string	(wymagane) Transport używany przez typ mediów. Możliwe wartości:[1] EMAIL (0), SMS (1), SCRIPT (2), WEBHOOK (4).
status	string	Czy typ mediów jest włączony. Możliwe wartości:[1] ENABLED (0, domyślnie), DISABLED (1)
max_sessions	integer	Maksymalna liczba alertów, które mogą być przetwarzane równolegle. Możliwe wartości dla SMS:[1] 1 (domyślnie). Możliwe wartości dla innych typów mediów:[1] 0-100 (gdzie 0 - bez ograniczeń).
attempts	integer	Maksymalna liczba prób wysłania alertu. Możliwe wartości:[1] 1-10 (domyślnie: 3).
attempt_interval	string	Odstęp (w sekundach lub z użyciem sufiksu czasu) między kolejnymi próbami ponowienia. Możliwe wartości:[1] 0-60s (domyślnie: 10s).
description	string	Opis typu mediów.
message_templates		Element główny dla szablonów wiadomości typu mediów.
event_source	string	(wymagane) Źródło zdarzenia. Możliwe wartości:[1] TRIGGERS (0), DISCOVERY (1), AUTOREGISTRATION (2), INTERNAL (3), SERVICE (4).
operation_mode	string	Tryb operacji. Możliwe wartości:[1] PROBLEM (0), RECOVERY (1), UPDATE (2).
subject	string	Temat wiadomości.
message	string	Treść wiadomości.

Note:
Zobacz także: Obiekt typu mediów (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Email

Poniższe dodatkowe elementy są eksportowane tylko dla typu mediów *Email*.

Element	Type	Description
provider	string	Dostawca poczty e-mail.
smtp_server	string	Serwer SMTP.
smtp_port	integer	Port serwera SMTP, z którym należy się połączyć. Domyślnie: 25.
smtp_helo	string	SMTP helo.
smtp_email	string	Adres e-mail, z którego będą wysyłane powiadomienia.
smtp_security	string	Poziom zabezpieczeń połączenia SMTP do użycia. Możliwe wartości:[1] NONE (0, domyślnie), STARTTLS (1), SSL_OR_TLS (2).
smtp_verify_host	string	Weryfikacja hosta SSL dla SMTP. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
smtp_verify_peer	string	Weryfikacja peer SSL dla SMTP. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
smtp_authentication	string	Metoda uwierzytelniania SMTP do użycia. Możliwe wartości:[1] NONE (0, domyślnie), PASSWORD (1), OAUTH (2).
username	string	Nazwa użytkownika.
password	string	Hasło uwierzytelniania.
redirection_url	string	URL frontend Zabbixa, na który następuje przekierowanie po autoryzacji OAuth.
client_id	string	Identyfikator klienta zarejestrowany w serwerze autoryzacji OAuth.
authorization_url	string	URL OAuth z parametrami do uzyskania tokenów dostępu i odświeżania.
token_url	string	URL OAuth do wymiany tokenu autoryzacji na tokeny dostępu i odświeżania.
message_format	string	Format wiadomości. Możliwe wartości:[1] TEXT (0), HTML (1, domyślnie).

Note:
Zobacz także: Obiekt typu mediów (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

SMS

Następujące dodatkowe elementy są eksportowane tylko dla typu mediów *SMS*.

Element	Type	Description
gsm_modem	string	(required) Nazwa urządzenia szeregowego modemu GSM.

Note:
Zobacz także: Media type object (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Skrypt

Poniższe dodatkowe elementy są eksportowane tylko dla typu mediów *Script*.

Element	Type	Description
nazwa skryptu	string	(wymagane) Nazwa skryptu.
parametry		Element główny dla parametrów skryptu.
sortorder	string	(wymagane) Kolejność parametrów skryptu przekazywanych do skryptu jako argumenty wiersza poleceń.
value	string	Wartość parametru skryptu.

Note:
Zobacz także: Obiekt typu mediów (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Webhook

Następujące dodatkowe elementy są eksportowane tylko dla typu mediów *Webhook*.

Element	Type	Description
script	string	Skrypt.
timeout	string	Limit czasu żądania HTTP skryptu Javascript. Possible values:[1] 1-60s (default: 30s).
process_tags	string	Czy przetwarzać zwrócone tagi. Possible values:[1] NO (0, default), YES (1).
show_event_menu	string	Wskazuje obecność wpisu w menu zdarzenia, jeśli makro {EVENT.TAGS.*} zostało pomyślnie rozwiązane w polach `event_menu_url` i `event_menu_name`. Possible values:[1] NO (0, default), YES (1).
event_menu_url	string	Adres URL wpisu menu zdarzenia. Obsługuje makro {EVENT.TAGS.*}.
event_menu_name	string	Nazwa wpisu menu zdarzenia. Obsługuje makro {EVENT.TAGS.*}.
parameters		Element nadrzędny dla parametrów typu mediów webhook.
name	string	(required) Nazwa parametru webhook.
value	string	Wartość parametru webhook.

Note:
Zobacz także: Obiekt typu mediów (odwołaj się do odpowiedniej właściwości o zgodnej nazwie).

Przypisy

[1] Wartości całkowite API w nawiasach, na przykład ENABLED (0), są podane wyłącznie jako informacja pomocnicza. Więcej informacji można znaleźć na powiązanej stronie obiektu API w pozycji tabeli lub na końcu każdej sekcji.

Pulpity

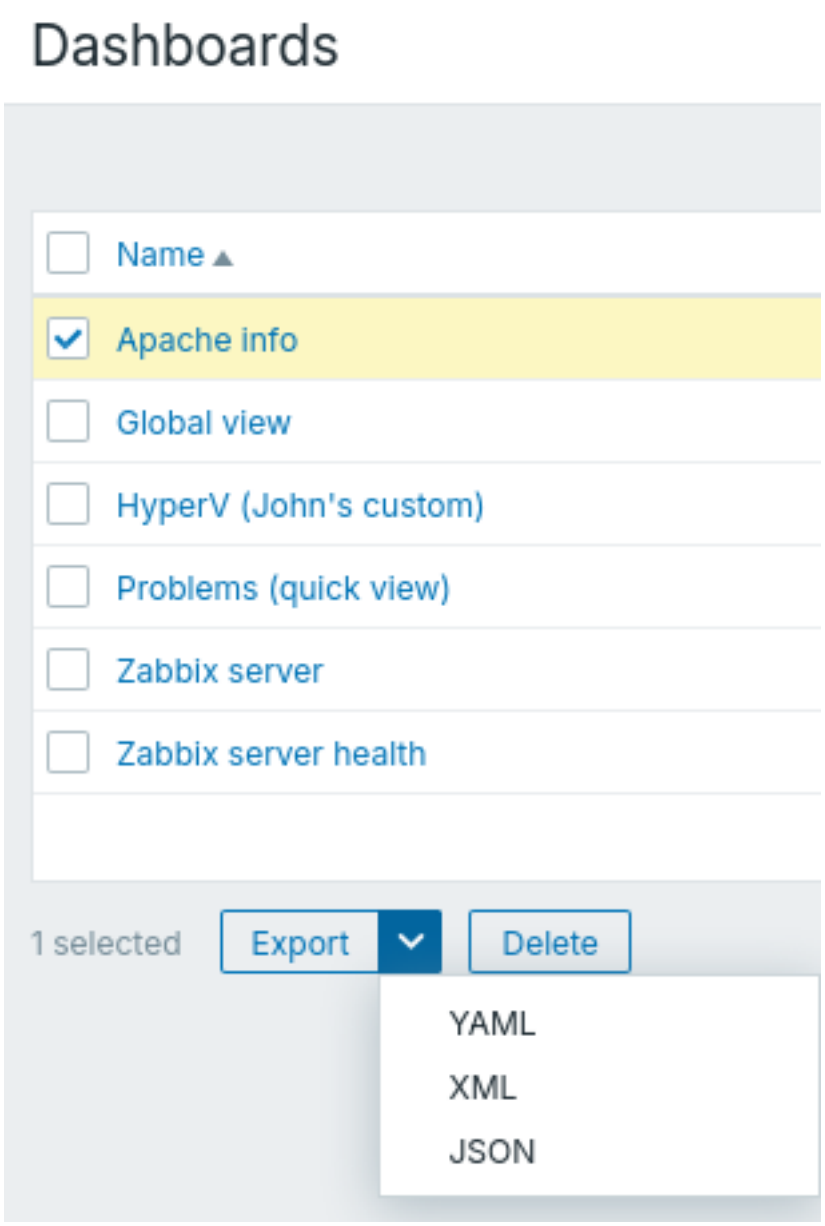
Przegląd

Pulpity nawigacyjne są eksportowane wraz ze wszystkimi powiązanymi obiektami i relacjami między obiektami.

Eksportowanie

Aby wyeksportować dashboardy, wykonaj następujące czynności:

1. Przejdź do *Dashboards* > *All dashboards*.
2. Zaznacz pola wyboru przy dashboardach, które chcesz wyeksportować.
3. Kliknij *Export* pod listą, aby wyeksportować w domyślnym formacie YAML, lub kliknij strzałkę, aby wybrać inny format eksportu.



W zależności od wybranego formatu dashboardy są eksportowane do lokalnego pliku o domyślnej nazwie:

- `zbx_export_dashboards.yaml` - w eksporcie YAML;
- `zbx_export_dashboards.xml` - w eksporcie XML;
- `zbx_export_dashboards.json` - w eksporcie JSON.

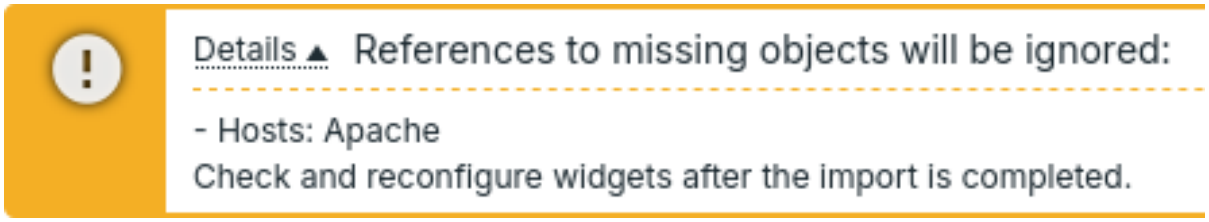
Importowanie

Aby zaimportować pulpity, wykonaj następujące czynności:

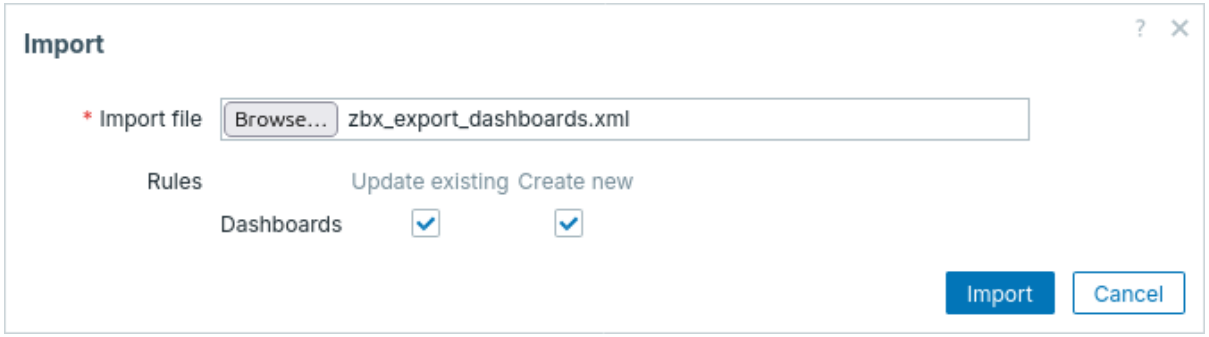
1. Przejdź do sekcji *Pulpity* > *Wszystkie pulpity*.
2. Kliknij przycisk *Importuj* w prawym górnym rogu. Przycisk jest nieaktywny, jeśli użytkownik nie ma uprawnień do *tworzenia i edytowania pulpitów*.

3. Wybierz plik importu.
4. Zaznacz pola wyboru reguł importu, które chcesz zastosować (zobacz poniżej).
5. Kliknij przycisk *Importuj* w prawym dolnym rogu formularza konfiguracji.

Jeśli plik importu odwołuje się do obiektów, które nie istnieją w systemie docelowym, zostanie wyświetlony komunikat ostrzegawczy. Jest to oczekiwane w przypadku widżetów odwołujących się do określonych hostów, pozycji lub innych obiektów. W takich przypadkach Zabbix umożliwia zaimportowanie pulpitu, usuwa brakujące odwołania i oznacza widżet jako nie w pełni skonfigurowany, jeśli dane są wymagane, albo usuwa kryteria filtrowania, jeśli pola są używane wyłącznie do filtrowania.



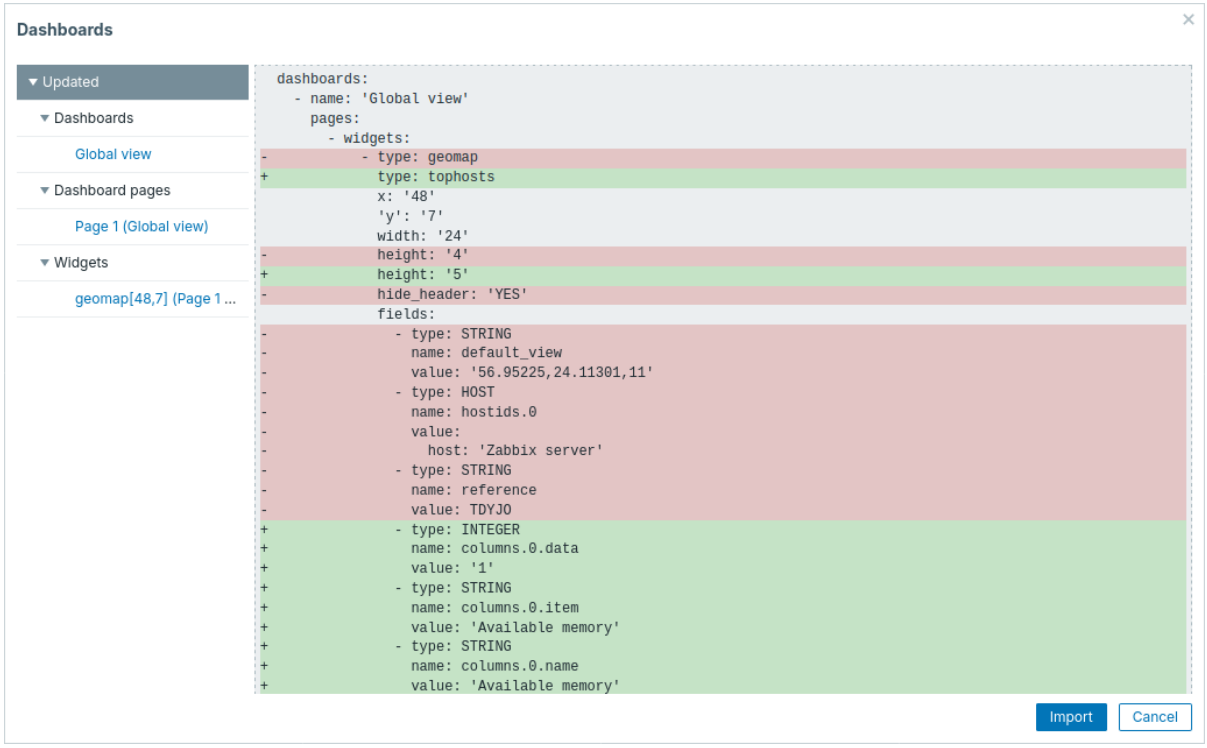
Ponowne kliknięcie przycisku *Importuj* spowoduje dodanie i/lub aktualizację pulpitu.



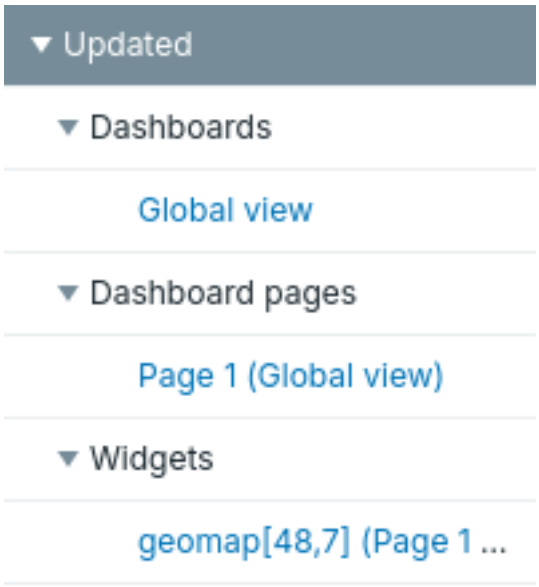
Reguły importu:

Reguła	Opis
Aktualizuj istniejące	Istniejące elementy zostaną zaktualizowane przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną zaktualizowane.
Utwórz nowe	Nowe elementy zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.

Na następnym ekranie będzie można wyświetlić zawartość importowanych pulpitów. Jeśli jest to nowy pulpit, wszystkie elementy zostaną wyświetlone na zielono. W przypadku aktualizowania istniejącego pulpitu nowe elementy pulpitu zostaną wyróżnione na zielono, usunięte elementy pulpitu zostaną wyróżnione na czerwono, a elementy, które nie uległy zmianie, zostaną wyświetlone na szarym tle.



Menu po lewej stronie umożliwia poruszanie się po liście zmian. Sekcja *Zaktualizowane* wyróżnia wszystkie zmiany wprowadzone w istniejących elementach pulpitu. Sekcja *Dodane* zawiera listę nowych elementów pulpitu. Elementy w każdej sekcji są pogrupowane według typu elementu; kliknij szarą strzałkę, aby rozwinąć lub zwinąć grupę elementów.



Przejrzyj zmiany pulpitu, a następnie kliknij przycisk *Importuj*, aby wykonać import pulpitu. W frontend zostanie wyświetlony komunikat o pomyślnym lub nieudanym imporcie.

Format eksportu

Eksport do 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: 'Current problems'
        '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: 'Problems by severity'
```

```yaml
        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: 'Local time'
        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: 'Server performance'
      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'
```

Eksportowane elementy

Eksportowane elementy są opisane w poniższej tabeli.

Element	Type	Description
name	string	(wymagane) Nazwa dashboardu.
pages		Element główny dla stron dashboardu
widgets		Element główny dla stron dashboardu

Note:
Zobacz także: Obiekt Dashboard (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Widget dashboardu

Element	Type	Description
type	string	(wymagane) Typ widgetu dashboardu. Możliwe wartości: actionlog - Dziennik zdarzeń; clock - Zegar; discovery - Status wykrywania; favgraphs - Ulubione wykresy; favmaps - Ulubione mapy; gauge - Wskaźnik; geomap - Geomapa; graph - Wykres (klasyczny); graphprototype - Prototyp wykresu; honeycomb - Plaster miodu; hostavail - Dostępność hosta; hostcard - Karta hosta; hostnavigator - Nawigator hostów; itemcard - Karta pozycji; itemhistory - Historia pozycji; itemnavigator - Nawigator pozycji; item - Wartość pozycji; map - Mapa; navtree - Drzewo nawigacji mapy; piechart - Wykres kołowy; problemhosts - Hosty z problemami; problems - Problemy; problemsbysv - Problemy według ważności; scatterplot - Wykres punktowy; slareport - Raport SLA; svggraph - Wykres; systeminfo - Informacje o systemie; tophosts - Najlepsze hosty; topitems - Najlepsze pozycje; toptriggers - Najlepsze wyzwalacze; trigover - Przegląd wyzwalaczy; url - URL; web - Monitorowanie WWW.
x	integer	Pozycja pozioma od lewej strony dashboardu. Możliwe wartości:[1] 0-71.
y	integer	Pozycja pionowa od górnej części dashboardu. Możliwe wartości:[1] 0-63.
name	string	Niestandardowa nazwa widgetu.
width	integer	Szerokość widgetu. Możliwe wartości:[1] 1-72.
height	integer	Wysokość widgetu. Możliwe wartości:[1] 1-64.
hide_header	string	Ukryj nagłówek widgetu. Możliwe wartości:[1] NO (0, domyślnie), YES (1).
fields		Element główny dla pól widgetu dashboardu.

Note:
Zobacz także: Widget dashboardu (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Pole widżetu dashboardu

Element	Type	Description
type	string	(wymagane) Typ pola widżetu. Dopuszczalne wartości:[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	(wymagane) Nazwa pola widżetu.
value	mixed	(wymagane) Wartość pola widżetu, zależnie od typu pola.
host	string	Host pozycji.
key	string	Klucz pozycji.

Note:
Zobacz także: Pole widżetu dashboardu (odwołanie do odpowiedniej właściwości o pasującej nazwie).

Przypisy

[1] Wartości całkowite API w nawiasach, na przykład ENABLED (0), są podane wyłącznie jako informacja pomocnicza. Więcej informacji można znaleźć na powiązanej stronie obiektu API w pozycji tabeli lub na końcu każdej sekcji.

Wyrażenia regularne

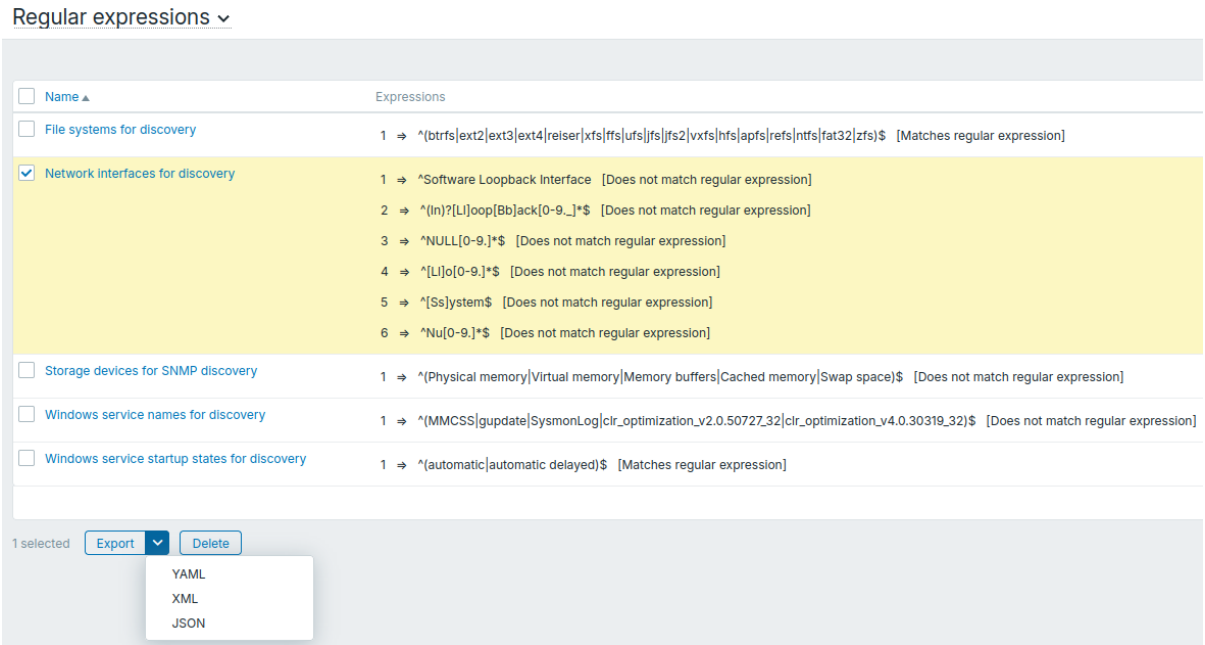
Przegląd

Wyrażenia regularne są eksportowane wraz ze wszystkimi powiązanymi obiektami i relacjami między obiektami.

Eksportowanie

Aby wyeksportować wyrażenia regularne, wykonaj następujące czynności:

1. Przejdź do *Administration* > *General* > *Regular expressions*.
2. Zaznacz pola wyboru przy wyrażeniach regularnych, które chcesz wyeksportować.
3. Kliknij *Export* pod listą, aby wyeksportować w domyślnym formacie YAML, lub kliknij strzałkę, aby wybrać inny format eksportu.



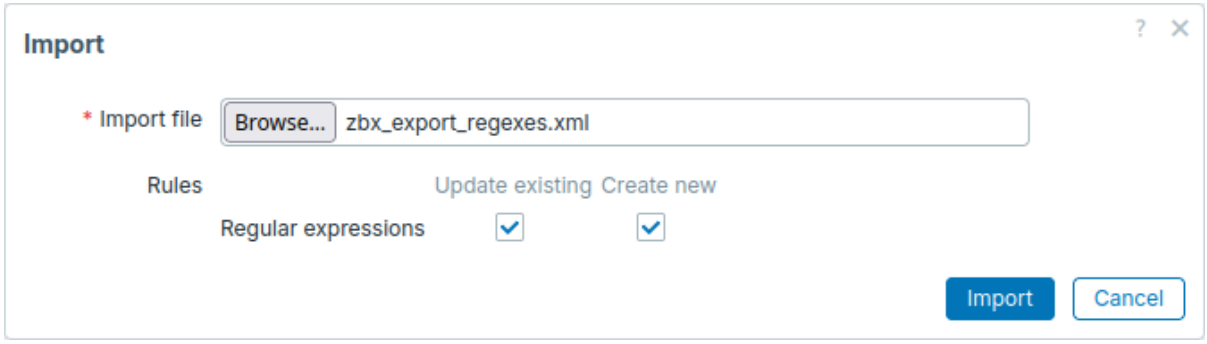
W zależności od wybranego formatu wyrażenia regularne są eksportowane do lokalnego pliku o domyślnej nazwie:

- `zbx_export_regexes.yaml` - w eksporcie YAML;
- `zbx_export_regexes.xml` - w eksporcie XML;
- `zbx_export_regexes.json` - w eksporcie JSON.

Importowanie

Aby zaimportować wyrażenia regularne, wykonaj następujące czynności:

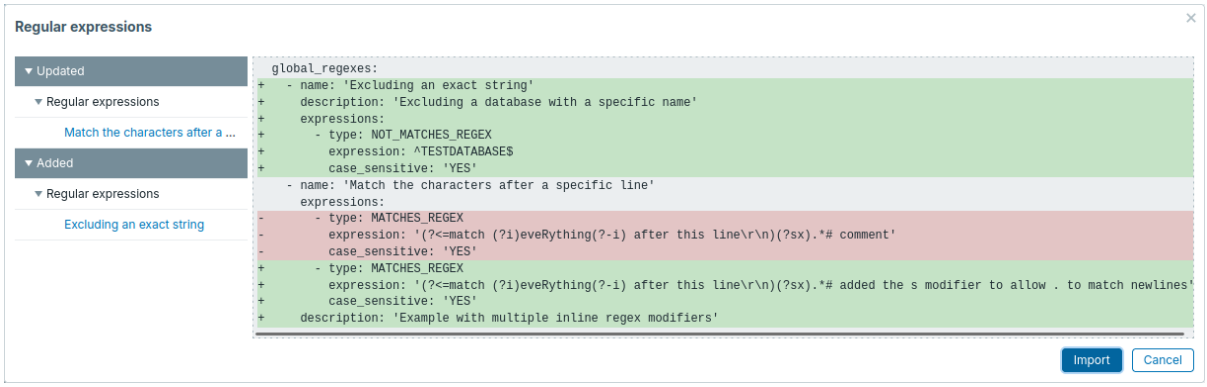
1. Przejdź do sekcji *Administration* > *General* > *Regular expressions*.
2. Kliknij przycisk *Import* w prawym górnym rogu.
3. Wybierz plik importu.
4. Zaznacz pola wyboru reguł importu, które chcesz zastosować (zobacz poniżej).
5. Kliknij przycisk *Import* w prawym dolnym rogu formularza konfiguracji.



Reguły importu:

Rule	Description
Update existing	Istniejące wyrażenia regularne zostaną zaktualizowane przy użyciu danych z pliku importu, całkowicie zastępując istniejące wyrażenia o tej samej nazwie. W przeciwnym razie nie zostaną zaktualizowane.
Create new	Nowe wyrażenia regularne zostaną utworzone przy użyciu danych z pliku importu. W przeciwnym razie nie zostaną utworzone.

Na następnym ekranie będzie można wyświetlić zawartość importowanych wyrażeń regularnych. Jeśli jest to nowe wyrażenie regularne, wszystkie dane zostaną wyświetlone na zielono. W przypadku aktualizowania istniejącego wyrażenia regularnego nowe dane wyrażenia regularnego zostaną wyróżnione na zielono, usunięte dane zostaną wyróżnione na czerwono, a niezmienione dane zostaną wyświetlone na szarym tle.



Menu po lewej stronie służy do poruszania się po liście zmian. Sekcja *Updated* wyróżnia wszystkie zmiany wprowadzone w istniejących wyrażeniach regularnych. Sekcja *Added* zawiera listę nowych wyrażeń regularnych. Kliknij strzałkę, aby rozwinąć lub zwinąć grupę wyrażeń regularnych.

Przejrzyj zmiany w wyrażeniach regularnych, a następnie kliknij przycisk *Import*, aby przeprowadzić import wyrażeń regularnych. W frontend zostanie wyświetlony komunikat o pomyślnym lub nieudanym imporcie.

Format eksportu

Eksport do YAML:

```
zabbix_export:
  version: '8.0'
  global_regexes:
    - name: 'File systems for discovery'
      description: 'Globalny wyrażenie regularne używane do wykrywania i identyfikowania systemów plików n
      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'
```

Eksportowane elementy

Eksportowane elementy wyrażeń regularnych:

Element	Type	Description
global_regexes	object	Element główny dla wyrażeń regularnych.
name	string	(wymagane) Nazwa wyrażenia regularnego.
description	text	Opis wyrażenia regularnego.
expressions	object	Element główny dla wyrażeń regularnych.
type	string	Typ wyrażenia regularnego.
expression	string	Wyrażenie regularne.
exp_delimiter	string	Separator wyrażenia regularnego; obecny, gdy typ to *Zawiera dowolny podciąg z listy*.
case_sensitive	string	Opcja rozróżniania wielkości liter dla wyrażenia regularnego.

Note:
Zobacz także: Obiekt wyrażenia regularnego (odwołaj się do odpowiedniej właściwości o pasującej nazwie).

Wyrażenia regularne

Przegląd Perl Compatible Regular Expressions (PCRE, PCRE2) są obsługiwane w Zabbix.

W Zabbix istnieją dwa sposoby używania wyrażeń regularnych:

- Ręczne wprowadzanie wyrażenia regularnego
- Używanie globalnego wyrażenia regularnego utworzonego w Zabbix

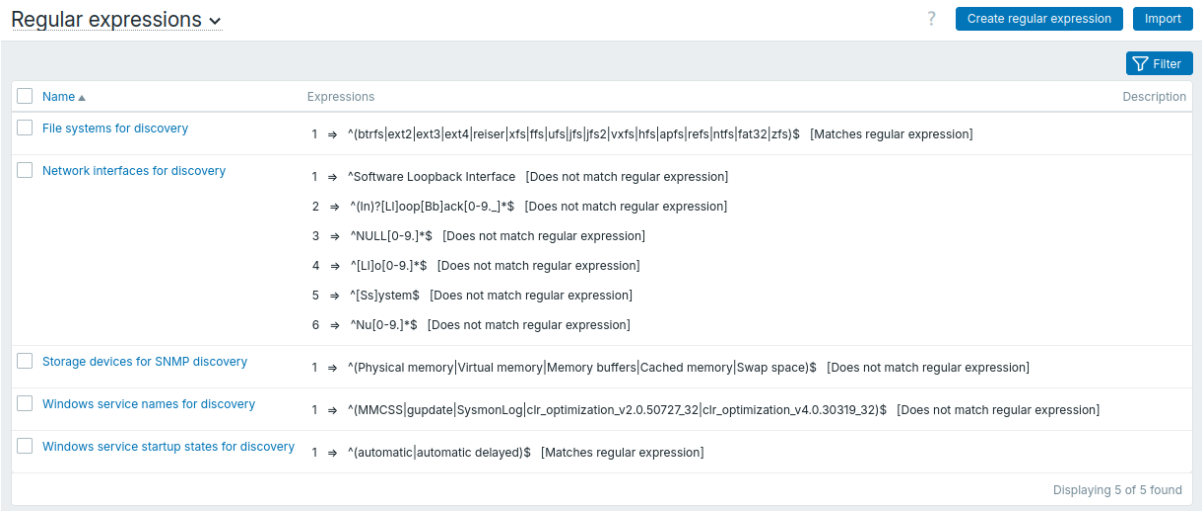
Wyrażenia regularne Możesz ręcznie wprowadzić wyrażenie regularne w obsługiwanych miejscach.
 Pamiętaj, że wyrażenie nie może zaczynać się od @, ponieważ ten symbol jest używany w Zabbix do odwoływania się do globalnych wyrażeń regularnych.

Attention:
Podczas używania wyrażeń regularnych może dojść do wyczerpania pamięci stosu. Więcej informacji znajdziesz w PCRE PERFORMANCE.

Pamiętaj, że w dopasowywaniu wielowierszowym kotwice ^ i $ odpowiadają odpowiednio początkowi i końcowi każdej linii, a nie początkowi i końcowi całego ciągu.

Zobacz także przykłady poprawnego escapowania w różnych kontekstach.

Globalne wyrażenia regularne Zabbix udostępnia zaawansowany edytor w frontendzie do tworzenia i testowania złożonych wyrażeń regularnych.
 Aby uzyskać do niego dostęp, przejdź do *Administration* > *General* > *Regular expressions*.
 Zabbix zawiera w domyślnym zestawie danych kilka predefiniowanych globalnych wyrażeń regularnych.

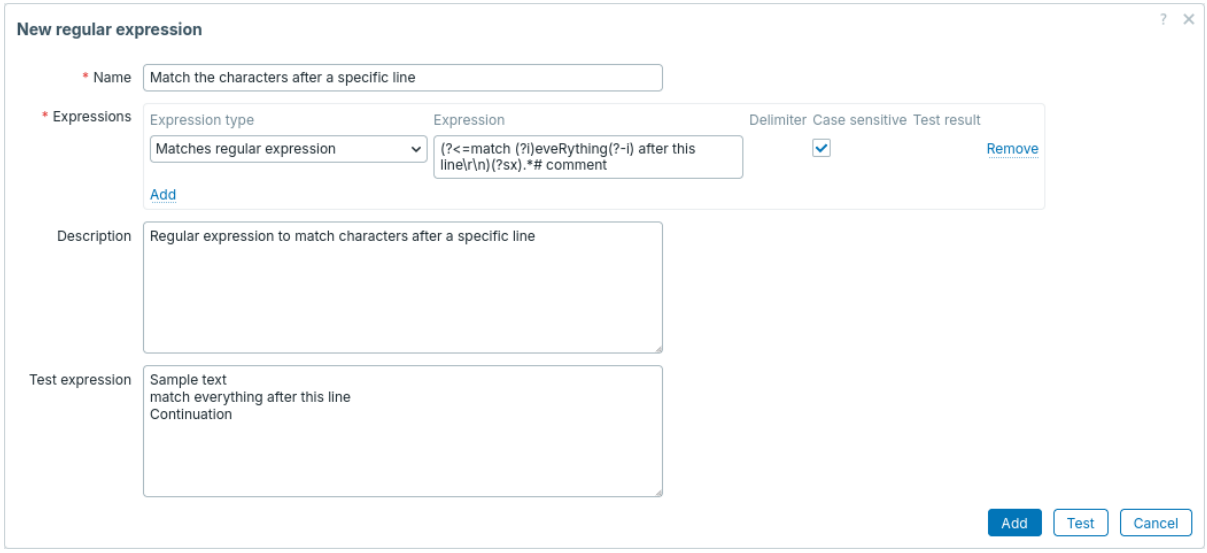


Możesz filtrować listę wyrażeń regularnych według nazwy i opisu, aby znaleźć potrzebne wyrażenia.

Konfiguracja

Aby utworzyć globalne wyrażenie regularne:

1. Kliknij *Utwórz wyrażenie regularne*.
2. Wprowadź parametry wyrażenia regularnego w formularzu.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa	Ustaw nazwę wyrażenia regularnego. Dozwolone są dowolne znaki Unicode.
Wyrażenia	Kliknij *Dodaj* w sekcji *Wyrażenia*, aby dodać nowe podwyrażenie.
Typ wyrażenia	Wybierz typ wyrażenia: **Zawiera ciąg znaków** - dopasowuje podciąg. **Zawiera dowolny podciąg z listy** - dopasowuje dowolny podciąg z listy rozdzielanej ogranicznikami. Lista rozdzielana ogranicznikami zawiera przecinek `,`, kropkę `.` lub ukośnik `/`. **Nie zawiera ciągu znaków** - dopasowuje dowolny ciąg z wyjątkiem podciągu. **Pasuje do wyrażenia regularnego** - dopasowuje wyrażenie regularne. **Nie pasuje do wyrażenia regularnego** - nie dopasowuje wyrażenia regularnego.
Wyrażenie	Wprowadź podciąg/wyrażenie regularne. Maksymalna długość: 2048 znaków.
Ogranicznik	Przecinek `,`, kropka `.` lub ukośnik `/` służący do oddzielania ciągów tekstowych w wyrażeniu regularnym. Ten parametr jest aktywny tylko wtedy, gdy wybrano typ wyrażenia *Zawiera dowolny podciąg z listy*.
Uwzględnia wielkość liter	Pole wyboru określające, czy wyrażenie regularne rozróżnia wielkość liter.
Opis	Wprowadź opis wyrażenia regularnego.
Testuj wyrażenie	Wartość używana do testowania wyrażenia regularnego.

Ukośnik `/` w wyrażeniu jest traktowany dosłownie, a nie jako ogranicznik. Umożliwia to zapisywanie wyrażeń zawierających ukośnik bez błędów.

Attention:
Nazwa niestandardowego wyrażenia regularnego w Zabbix może zawierać przecinki, spacje itp. W przypadkach, w których może to prowadzić do błędnej interpretacji podczas odwoływania się do nazwy (na przykład przecinek w parametrze klucza pozycji), całe odwołanie można ująć w cudzysłowy, na przykład: *"@My custom regexp for purpose1, purpose2"*.
Nazwy wyrażeń regularnych nie mogą być ujmowane w cudzysłowy w innych miejscach (na przykład we właściwościach reguły LLD).

Po utworzeniu wyrażenia regularnego można go używać w kilku miejscach w frontend, odwołując się do jego nazwy poprzedzonej znakiem `@`, na przykład `@mycustomregexp`.

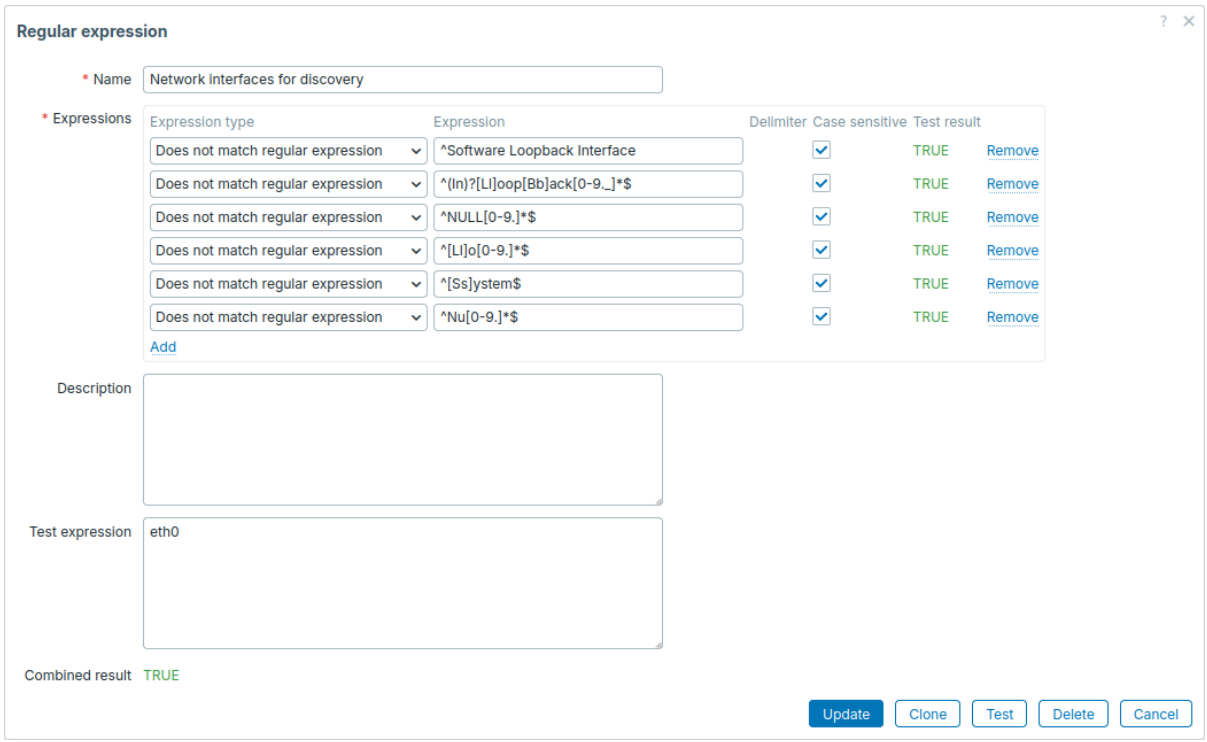
Przyciski formularza

Przyciski u dołu formularza umożliwiają wykonywanie różnych operacji:

Button	Description
Add	Dodaj wyrażenie regularne. Ten przycisk jest dostępny tylko dla nowych wyrażeń regularnych.
Update	Zaktualizuj właściwości wyrażenia regularnego.
Clone	Utwórz kolejne wyrażenie regularne na podstawie właściwości bieżącego wyrażenia regularnego.
Test	Przetestuj wyrażenie regularne i jego podwyrażenia względem podanej wartości.
Delete	Usuń wyrażenie regularne.
Cancel	Anuluj edycję właściwości wyrażenia regularnego.

Testowanie wyrażenia regularnego

Możesz przetestować wyrażenie regularne oraz jego podwyrażenia, wpisując wartość w polu *Test expression* i klikając przycisk *Test button*.



W kolumnie *Test result* wyświetlany jest status każdego podwyrażenia. Pole *Combined result* wyświetla ogólny status wyrażenia regularnego. Jeśli zdefiniowano wiele podwyrażeń, Zabbix używa operatora logicznego AND do obliczenia *Combined result*; to znaczy, jeśli co najmniej jeden wynik ma wartość `False`, *Combined result* również ma wartość `False`.

Domyślne globalne wyrażenia regularne

Domyślny zestaw danych zawiera następujące predefiniowane globalne wyrażenia regularne:

Nazwa	Wyrażenie	Dopasowania
Systemy plików do wykrywania	`^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32|zfs)$`	”btrfs” lub ”ext2” lub ”ext3” lub ”ext4” lub ”reiser” lub ”xfs” lub ”ffs” lub ”ufs” lub ”jfs” lub ”jfs2” lub ”vxfs” lub ”hfs” lub ”apfs” lub ”refs” lub ”ntfs” lub ”fat32” lub ”zfs”.
Interfejsy sieciowe do wykrywania	`^Software Loopback Interface`	Ciągi zaczynające się od ”Software Loopback Interface”.
	`^lo$`	”lo”.
	`^(In)?[Ll]oop[Bb]ack[0-9._]*$`	Ciągi opcjonalnie zaczynające się od ”In”, po których następuje ”L” lub ”l”, następnie ”oop”, potem ”B” lub ”b”, następnie ”ack”, a opcjonalnie dowolna liczba cyfr, kropek lub podkreśleń.
	`^NULL[0-9.]*$`	Ciągi zaczynające się od ”NULL”, opcjonalnie po których następuje dowolna liczba cyfr lub kropek.

Nazwa	Wyrażenie	Dopasowania
	`^[Ll]o[0-9.]*$`	Ciągi zaczynające się od ”Lo” lub ”lo” i opcjonalnie po których następuje dowolna liczba cyfr lub kropek.
	`^[Ss]ystem$`	”System” lub ”system”.
	`^Nu[0-9.]*$`	Ciągi zaczynające się od ”Nu”, opcjonalnie po których następuje dowolna liczba cyfr lub kropek.
Urządzenia pamięci masowej do wykrywania SNMP	`^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$`	”Physical memory” lub ”Virtual memory” lub ”Memory buffers” lub ”Cached memory” lub ”Swap space”.
Nazwy usług Windows do wykrywania	`^(MMCSS|gupdate|SysmonLog|clr_optimization_v2.0.50727_32|clr_optimization_v4.0.30319_32)$`	”MMCSS”, ”gupdate” lub ”SysmonLog” lub ciągi pasujące do ”clr_optimization_v2.0.50727_32” lub ”clr_optimization_v4.0.30319_32”, przy czym każda kropka może zostać zastąpiona dowolnym znakiem z wyjątkiem znaku nowej linii.
Stany uruchamiania usług Windows do wykrywania	`^(automatic|automatic delayed)$`	”automatic” lub ”automatic delayed”.

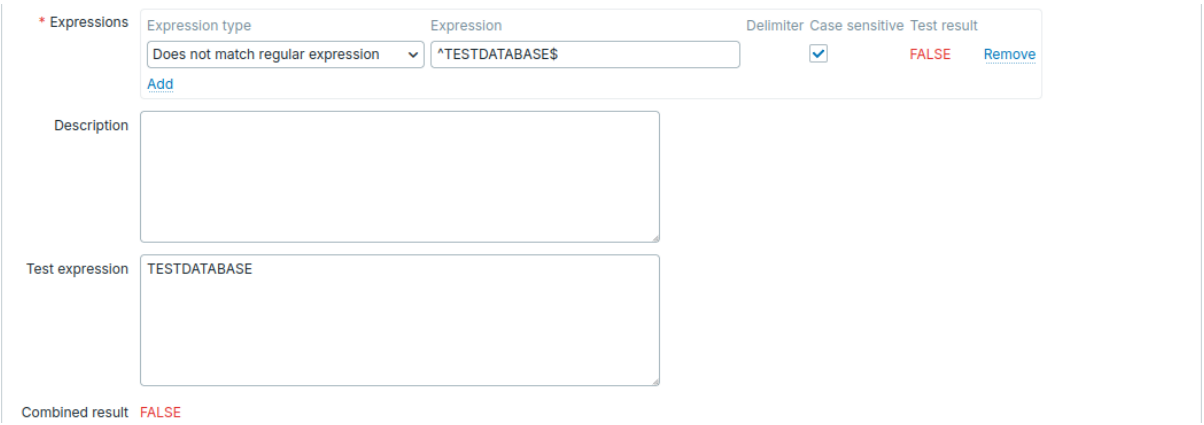
Importowanie/eksportowanie wyrażenia regularnego

Możesz importować lub eksportować globalne wyrażenia regularne, jak opisano na stronie konfiguracji eksportu/importu Wyrażenia regularne.

Przykłady Przykład wykluczania dokładnego ciągu

Użyj następującego wyrażenia w wykrywaniu niskiego poziomu, aby wykrywać bazy danych z wyjątkiem bazy danych o określonej nazwie:

```
^TESTDATABASE$
```

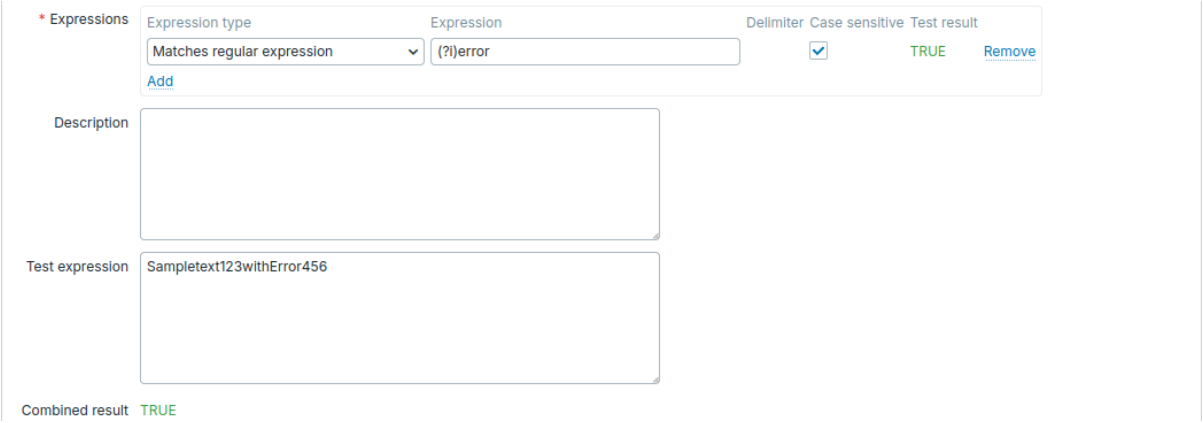


Zarówno *Test result*, jak i *Combined result* wyświetlają FALSE: nazwa pasuje do ”TESTDATABASE”, więc warunek *Does not match regular expression* nie jest spełniony.

Przykład z wbudowanym modyfikatorem wyrażenia regularnego dla dopasowania bez uwzględniania wielkości liter

Użyj następującego wyrażenia regularnego, które zawiera wbudowany modyfikator `(?i)`, aby wykonać dopasowanie ciągu znaków ”error” bez uwzględniania wielkości liter:

```
(?i)error
```



Zarówno *Test result*, jak i *Combined result* wyświetlają TRUE: znaki "error" są dopasowywane nawet przy zaznaczonej opcji *Case sensitive*, ponieważ wbudowany modyfikator `(?i)` sprawia, że dopasowanie nie uwzględnia wielkości liter.

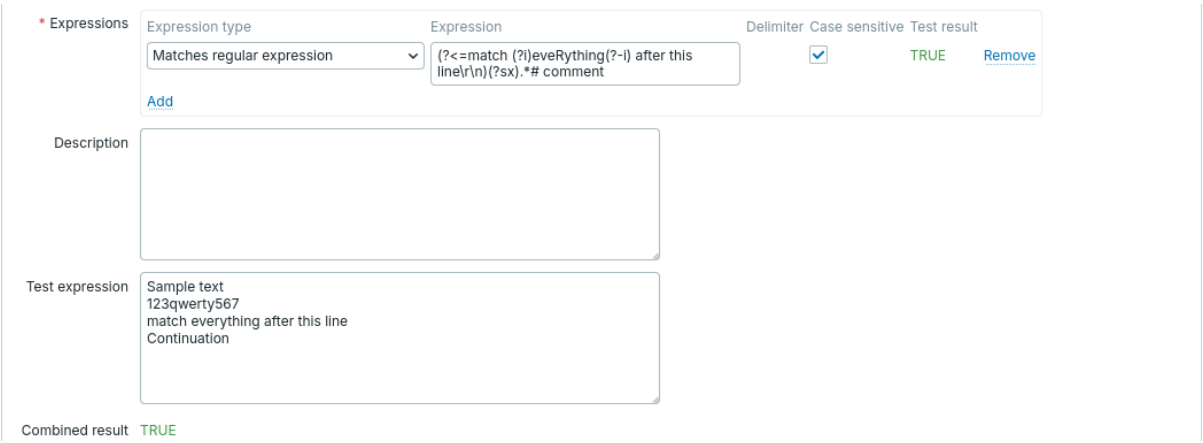
Przykład z wieloma modyfikatorami wyrażeń regularnych inline

Użyj następującego wyrażenia regularnego, które zawiera wiele modyfikatorów inline, aby dopasować znaki po określonej linii:

```
(?<=match (?i)eveRything(?-i) after this line\r\n)(?sx).*# comment
```

gdzie:

- `(?i)` - włącza dopasowywanie bez rozróżniania wielkości liter;
- `(?-i)` - wyłącza dopasowywanie bez rozróżniania wielkości liter;
- `(?s)` - włącza tryb dotall, umożliwiając `.` dopasowanie znaków nowej linii;
- `(?x)` - włącza tryb rozszerzony, pozwalając ignorować białe znaki i komentarze w wyrażeniu regularnym.



Zarówno *Test result*, jak i *Combined result* wyświetlają TRUE: znaki następujące po określonej linii są dopasowane.

Attention:
Modyfikatora *g* nie można określić inline. Listę dostępnych modyfikatorów można znaleźć na stronie podręcznika pcresyntax. Więcej informacji na temat składni PCRE można znaleźć w dokumentacji HTML PCRE.

Obsługa wyrażeń regularnych według lokalizacji

Location		Regular expression	Global regular expression	Multiline matching	Comments
Pozycje agenta					
	eventlog[]	Yes	Yes	Yes	Parametry `regexp`, `severity`, `source`, `eventid`
	eventlog.count[]				Parametry `regexp`, `severity`, `source`, `eventid`
	log[]				Parametr `regexp`
	log.count[]				

Location		Regular expression	Global regular expression	Multiline matching	Comments
	logrt[]		Yes/No		Parametr `regexp` obsługuje oba warianty, parametr `file_regexp` obsługuje tylko wyrażenia nieglobalne
	logrt.count[]				
	proc.cpu.util[] proc.get[] proc.mem[] proc.num[]		No	No	Parametr `cmdline`
	sensor[]				Parametry `device` i `sensor` w Linux 2.4
	system.hw.macaddr[]				Parametr `interface`
	system.sw.packages[]				Parametr `regexp`
	system.sw.packages.get[]				Parametr `regexp`
	vfs.dir.count[]				Parametry `regex_incl`, `regex_excl`, `regex_excl_dir`
	vfs.dir.get[]				Parametry `regex_incl`, `regex_excl`, `regex_excl_dir`
	vfs.dir.size[]				Parametry `regex_incl`, `regex_excl`, `regex_excl_dir`
	vfs.file.regexp[] vfs.file.regmatch[] web.page.regexp[]			Yes	Parametr `regexp`
Trapy SNMP					
	snmptrap[]	Yes	Yes	No	Parametr `regexp`
Wstępne przetwarzanie wartości		Yes	No	No	Pole `pattern`
Funkcje dla wyzwalaczy/pozycji obliczanych					
	count()	Yes	Yes	Yes	Parametr `pattern`, jeśli parametr `operator` ma wartość *regexp* lub *iregexp*
	countunique()	Yes	Yes		
	find()	Yes	Yes		
	logeventid()	Yes	Yes	No	Parametr `pattern`
	logsource()				
Wykrywanie niskiego poziomu					
	Filters	Yes	Yes	No	Pole *Regular expression*
	Overrides	Yes	No		W opcjach *matches*, *does not match* dla warunków *Operation*
Warunki akcji		Yes	No	No	W opcjach *matches*, *does not match* dla warunków automatycznej rejestracji *Host name* i *Host metadata*
Skrypty		Yes	Yes	No	Pole *Input validation rule*
Monitorowanie WWW		Yes	No	Yes	*Variables* z prefiksem **regex:** Pole *Required string*

Location		Regular expression	Global regular expression	Multiline matching	Comments
Kontekst makra użytkownika		Yes	No	No	W kontekście makra z prefiksem **regex:**
Funkcje makr					
	regsub() iregsub()	Yes	No	No	Parametr `pattern`
Wskaźniki łączy na mapach		Yes	No	No	Pole *Pattern* (dla pozycji tekstowych)
Mapowanie ikon		Yes	Yes	No	Pole *Expression*
Mapowanie wartości		Yes	No	No	Pole *Value*, jeśli typ mapowania to `regexp`
Parametry konfiguracji agenta					
	AllowKeyRegexp DenyKeyRegexp	Yes	No	No	Wzorzec

Zdarzenia i rozwiązanie

Omówienie W Zabbix generowanych jest kilka typów zdarzeń:

- zdarzenia wyzwalaczy - za każdym razem, gdy wyzwalacz zmienia swój status (*OK→PROBLEM→OK*)
- zdarzenia usług - za każdym razem, gdy usługa zmienia swój status (*OK→PROBLEM→OK*)
- zdarzenia wykrywania - gdy hosty lub usługi zostaną wykryte
- zdarzenia automatycznej rejestracji - gdy aktywne agent są automatycznie rejestrowane przez serwer
- zdarzenia wewnętrzne - gdy pozycja/reguła odkrywania niskiego poziomu staje się niewspierana lub wyzwalacz przechodzi w nieznany stan

Zdarzenia są oznaczone znacznikiem czasu i mogą stanowić podstawę działań, takich jak wysyłanie wiadomości e-mail z powiadomieniem, tworzenie zgłoszeń lub wykonywanie zdalnych poleceń.

Aby wyświetlić szczegóły zdarzenia w frontend, przejdź do *Monitoring* > *Problems*. Tam możesz kliknąć datę i godzinę zdarzenia, aby otworzyć stronę szczegółów zdarzenia, gdzie można zobaczyć დამატკowe informacje o zdarzeniu, w tym to, czy zostało ono wstrzymane z powodu konserwacji lub ręcznie wstrzymane przez użytkownika.

Rozwiązywanie problemów Zdarzenia problemów utworzone przez wyzwalacze są rozwiązywane automatycznie, jeśli warunek, który spowodował problem, już nie istnieje.

Aby zwiększyć elastyczność i wygodę zarządzania zdarzeniami problemów, dostępne są następujące opcje:

- Korelacja zdarzeń
- Potwierdzanie i tłumienie zdarzeń
- Ręczne zamykanie problemów
- Okresy konserwacji - definiują okresy, w których problemy są tłumione
- Tagowanie - oznaczanie problemów według wspólnej cechy

Zdarzenia wyzwalacza Zmiana stanu wyzwalacza jest najczęstszym i najważniejszym źródłem zdarzeń. Za każdym razem, gdy wyzwalacz zmienia swój stan, generowane jest zdarzenie. Zdarzenie zawiera szczegóły zmiany stanu wyzwalacza - kiedy nastąpiła i jaki jest nowy stan.

Wyzwalacze tworzą dwa typy zdarzeń - `Problem` i `OK`.

Zdarzenia problemowe

Zdarzenie problemowe jest tworzone:

- gdy wyrażenie wyzwalacza zostanie ocenione jako TRUE, jeśli wyzwalacz znajduje się w stanie OK;
- za każdym razem, gdy wyrażenie wyzwalacza zostanie ocenione jako TRUE, jeśli dla wyzwalacza włączono generowanie wielu zdarzeń problemowych.

Zdarzenia OK

Zdarzenie OK zamyka powiązane zdarzenie(-a) problemu i może być utworzone przez 3 komponenty:

- wyzwalacze - na podstawie ustawień „OK event generation" i „OK event closes";
- korelacja zdarzeń
- menedżer zadań - gdy zdarzenie jest zamykane ręcznie

Wyzwalacze mają ustawienie „OK event generation", które kontroluje sposób generowania zdarzeń OK:

- *Expression* - zdarzenie OK jest generowane dla wyzwalacza w stanie problemu, gdy jego wyrażenie przyjmuje wartość FALSE. Jest to najprostsze ustawienie, domyślnie włączone.
- *Recovery expression* - zdarzenie OK jest generowane dla wyzwalacza w stanie problemu, gdy jego wyrażenie przyjmuje wartość FALSE, a wyrażenie odzyskiwania przyjmuje wartość TRUE. Można tego użyć, jeśli kryteria odzyskiwania wyzwalacza różnią się od kryteriów problemu.
- *None* - zdarzenie OK nigdy nie jest generowane. Można tego użyć w połączeniu z wielokrotnym generowaniem zdarzeń problemu, aby po prostu wysłać powiadomienie, gdy coś się wydarzy.

Dodatkowo wyzwalacze mają ustawienie „OK event closes", które kontroluje, które zdarzenia problemu są zamykane:

- *All problems* - zdarzenie OK zamknie wszystkie otwarte problemy utworzone przez wyzwalacz
- *All problems if tag values match* - zdarzenie OK zamknie otwarte problemy utworzone przez wyzwalacz i mające co najmniej jedną pasującą wartość tagu. Tag jest zdefiniowany przez ustawienie wyzwalacza „Tag for matching". Jeśli nie ma żadnych zdarzeń problemu do zamknięcia, zdarzenie OK nie jest generowane. Jest to często nazywane korelacją zdarzeń na poziomie wyzwalacza.

Korelacja zdarzeń

Korelacja zdarzeń (nazywana również globalną korelacją zdarzeń) to sposób konfigurowania niestandardowych reguł zamykania zdarzeń (skutkujących wygenerowaniem zdarzenia OK).

Reguły określają sposób parowania nowych zdarzeń problemowych z istniejącymi zdarzeniami problemowymi, umożliwiając zamknięcie nowego zdarzenia lub dopasowanych zdarzeń przez wygenerowanie odpowiednich zdarzeń OK.

Korelację zdarzeń należy jednak konfigurować bardzo ostrożnie, ponieważ może ona negatywnie wpłynąć na wydajność przetwarzania zdarzeń lub, w przypadku nieprawidłowej konfiguracji, zamknąć więcej zdarzeń, niż zamierzono (w najgorszym przypadku mogą zostać zamknięte nawet wszystkie zdarzenia problemowe). Kilka wskazówek dotyczących konfiguracji:

1. zawsze ograniczaj zakres korelacji, ustawiając unikatowy znacznik dla zdarzenia sterującego (zdarzenia parowanego ze starymi zdarzeniami), i używaj warunku korelacji „new event tag"
2. podczas korzystania z operacji „close old event" nie zapominaj o dodaniu warunku opartego na starym zdarzeniu, ponieważ w przeciwnym razie wszystkie istniejące problemy mogą zostać zamknięte
3. unikaj używania typowych nazw znaczników używanych przez różne konfiguracje korelacji

Menedżer zadań

Jeśli ustawienie „Allow manual close" jest włączone dla wyzwalacza, możliwe jest ręczne zamykanie zdarzeń problemu generowanych przez wyzwalacz. Wykonuje się to w frontend podczas aktualizowania problemu. Zdarzenie nie jest zamykane bezpośrednio - zamiast tego tworzone jest zadanie „close event", które wkrótce jest obsługiwane przez menedżer zadań. Menedżer zadań wygeneruje odpowiadające mu zdarzenie OK, a zdarzenie problemu zostanie zamknięte.

Zdarzenia usługi Zdarzenia usługi są generowane tylko wtedy, gdy akcje usługi dla tych zdarzeń są włączone. W takim przypadku każda zmiana stanu usługi tworzy nowe zdarzenie:

- Zdarzenie problemu - gdy stan usługi zmienia się z `OK` na `Problem`
- Zdarzenie OK - gdy stan usługi zmienia się z `Problem` na `OK`

Zdarzenie zawiera szczegóły zmiany stanu usługi - kiedy nastąpiła i jaki jest nowy stan.

Zdarzenia wykrywania Zabbix okresowo skanuje zakresy adresów IP zdefiniowane w regułach wykrywania sieci. Częstotliwość sprawdzania można skonfigurować osobno dla każdej reguły. Gdy host lub usługa zostanie wykryta, generowane jest zdarzenie wykrywania (lub kilka zdarzeń).

Zabbix generuje następujące zdarzenia:

Zdarzenie	Kiedy generowane
Service Up	Za każdym razem, gdy Zabbix wykryje aktywną usługę.
Service Down	Za każdym razem, gdy Zabbix nie może wykryć usługi.
Host Up	Jeśli co najmniej jedna z usług jest UP dla danego adresu IP.
Host Down	Jeśli wszystkie usługi nie odpowiadają.
Service Discovered	Jeśli usługa powróci po przestoju lub zostanie wykryta po raz pierwszy.
Service Lost	Jeśli usługa zostanie utracona po wcześniejszym stanie UP.
Host Discovered	Jeśli host powróci po przestoju lub zostanie wykryty po raz pierwszy.
Host Lost	Jeśli host zostanie utracony po wcześniejszym stanie UP.

Zdarzenia automatycznej rejestracji aktywnego agenta Automatyczna rejestracja aktywnego agenta tworzy zdarzenia w Zabbix.

Jeśli jest skonfigurowana, zdarzenie automatycznej rejestracji aktywnego agenta jest tworzone, gdy wcześniej nieznany aktywny agent prosi o sprawdzenia lub gdy metadane hosta uległy zmianie. Serwer dodaje nowo automatycznie zarejestrowany host, używając otrzymanego adresu IP i portu agenta.

Więcej informacji można znaleźć na stronie automatycznej rejestracji aktywnego agenta.

Zdarzenia wewnętrzne Zdarzenia wewnętrzne występują, gdy:

- pozycja zmienia stan z 'normalny' na 'unsupported'
- pozycja zmienia stan z 'unsupported' na 'normalny'
- reguła wykrywania niskiego poziomu zmienia stan z 'normalny' na 'unsupported'
- reguła wykrywania niskiego poziomu zmienia stan z 'unsupported' na 'normalny'
- wyzwalacz zmienia stan z 'normalny' na 'unknown'
- wyzwalacz zmienia stan z 'unknown' na 'normalny'

Celem wprowadzenia zdarzeń wewnętrznych jest umożliwienie użytkownikom otrzymywania powiadomień, gdy wystąpi dowolne zdarzenie wewnętrzne, na przykład gdy pozycja staje się nieobsługiwana i przestaje zbierać dane.

Zdarzenia wewnętrzne są tworzone tylko wtedy, gdy włączone są akcje wewnętrzne dla tych zdarzeń. Aby zatrzymać generowanie zdarzeń wewnętrznych (na przykład dla pozycji, które stają się nieobsługiwane), wyłącz wszystkie akcje dla zdarzeń wewnętrznych w *Alerts* > *Actions* > *Internal actions*.

Note:
Jeśli akcje wewnętrzne są wyłączone, a obiekt znajduje się w stanie 'unsupported', zdarzenie odzyskania dla tego obiektu nadal zostanie utworzone.
Jeśli akcje wewnętrzne są włączone, a obiekt znajduje się w stanie 'unsupported', zdarzenie odzyskania dla tego obiektu zostanie utworzone, mimo że dla obiektu nie zostało utworzone 'problem event'.

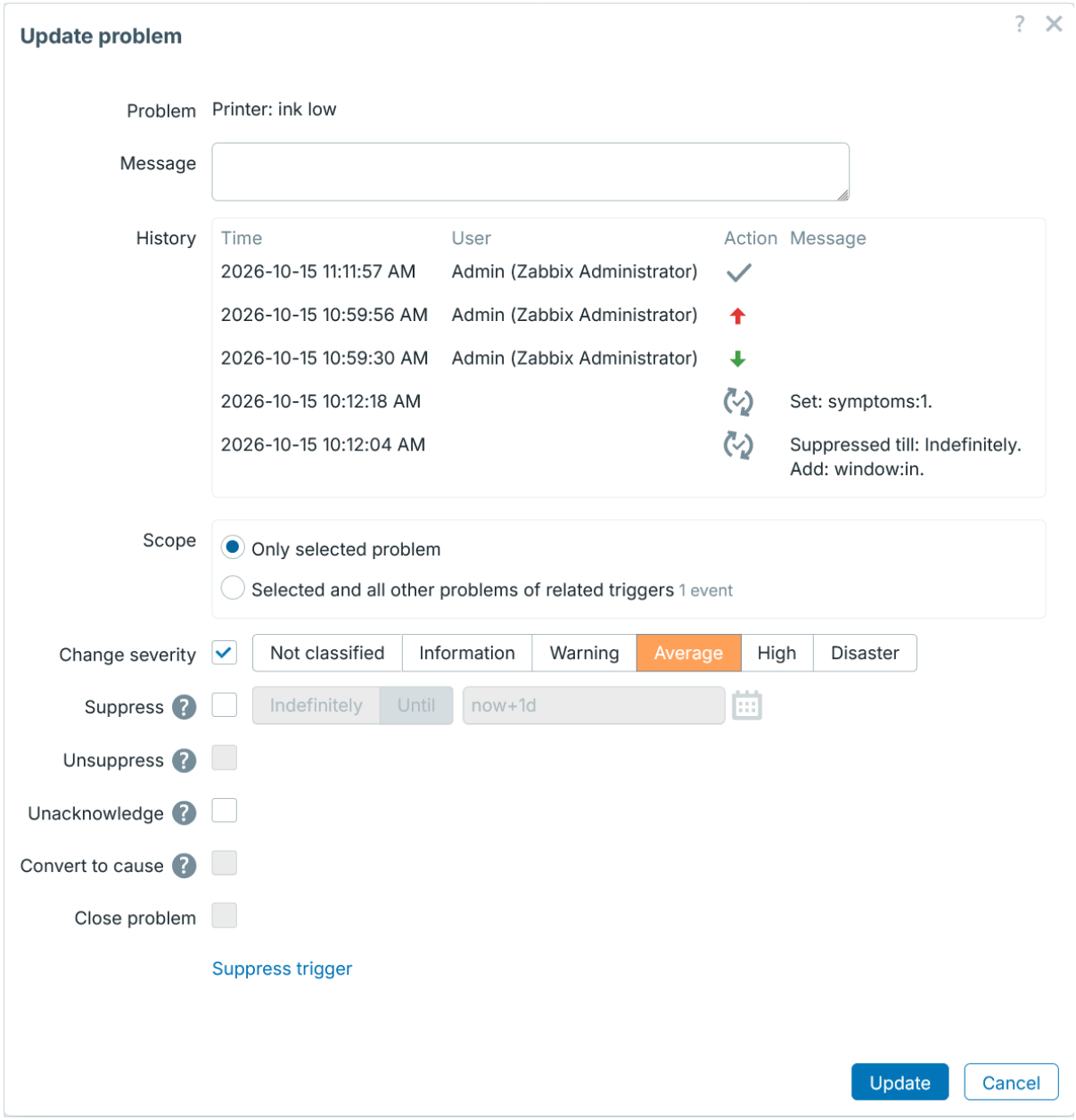
Zobacz także: Otrzymywanie powiadomień o pozycjach nieobsługiwanych

Potwierdzanie i tłumienie problemów

Zdarzenia problemów w Zabbix mogą być potwierdzane lub tłumione przez użytkowników.

Aktualizowanie problemów Okno podręczne aktualizacji problemu umożliwia:

- Dodanie komentarza do problemu.
- Wyświetlenie dotychczasowych komentarzy i działań.
- Zmianę ważności problemu.
- Wyciszenie/anulowanie wyciszenia problemu.
- Potwierdzenie/anulowanie potwierdzenia problemu.
- Zmianę problemu objawowego na problem przyczynowy.
- Ręczne zamknięcie problemu.
- Utworzenie doraźnego okresu konserwacji dla wyzwalacza.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Problem	Jeśli wybrano tylko jeden problem, wyświetlana jest nazwa problemu. Jeśli wybrano kilka problemów, wyświetlany jest komunikat *Wybrano N problemów*.
Message	Wprowadź tekst komentarza do problemu (maksymalnie 2048 znaków).
History	Na liście wyświetlane są wcześniejsze działania i komentarze dotyczące problemu, wraz z informacjami o czasie i użytkowniku. Znaczenie ikon używanych do oznaczania działań użytkownika opisano na stronie szczegółów zdarzenia. Należy pamiętać, że historia jest wyświetlana tylko wtedy, gdy do aktualizacji wybrano jeden problem.
Scope	Zdefiniuj zakres działań, takich jak zmiana ważności, potwierdzanie lub ręczne zamykanie problemów: **Tylko wybrany problem** - wpłynie tylko na to zdarzenie **Wybrany problem i wszystkie pozostałe problemy powiązanych wyzwalaczy** - w przypadku potwierdzania/zamykania problemu wpłynie na to zdarzenie oraz wszystkie pozostałe problemy, które nie zostały dotychczas potwierdzone/zamknięte. Jeśli zakres obejmuje problemy już potwierdzone lub zamknięte, nie zostaną one ponownie potwierdzone/zamknięte. Z drugiej strony liczba operacji dodawania komunikatów i zmiany ważności nie jest ograniczona.

Parametr	Opis
Change severity	Zaznacz pole wyboru i kliknij przycisk ważności, aby zaktualizować ważność problemu. Pole wyboru zmiany ważności jest dostępne, jeśli dla co najmniej jednego z wybranych problemów istnieją uprawnienia do odczytu i zapisu. Po kliknięciu przycisku *Update* zostaną zaktualizowane tylko problemy, dla których dostępny jest odczyt i zapis. Jeśli dla żadnego z wybranych wyzwalaczy nie istnieją uprawnienia do odczytu i zapisu, pole wyboru jest wyłączone.
Suppress	Zaznacz pole wyboru, aby wyciszyć problem: **Bezterminowo** - wyciszenie bezterminowe **Do** - wyciszenie do określonego czasu. Obsługiwane są zarówno bezwzględne, jak i względne formaty czasu, na przykład: `now+1d` - przez jeden dzień od teraz (domyślnie) `now/w` - do końca bieżącego tygodnia `2022-05-28 12:00:00` - do określonej daty/godziny Należy pamiętać, że prosty okres (np. `1d`, `1w`) nie jest obsługiwany. Dostępność tej opcji zależy od ustawień roli użytkownika „Suppress problems". Zobacz także: Wyciszanie problemów
Unsuppress	Zaznacz pole wyboru, aby anulować wyciszenie problemu. To pole wyboru jest aktywne tylko wtedy, gdy co najmniej jeden z wybranych problemów jest wyciszony. Dostępność tej opcji zależy od ustawień roli użytkownika „Suppress problems".
Acknowledge	Zaznacz pole wyboru, aby potwierdzić problem. To pole wyboru jest dostępne, jeśli wśród wybranych problemów znajduje się co najmniej jeden niepotwierdzony problem. Nie można dodać kolejnego potwierdzenia do już potwierdzonego problemu (można jednak dodać kolejny komentarz).
Unacknowledge	Zaznacz pole wyboru, aby anulować potwierdzenie problemu. To pole wyboru jest dostępne, jeśli wśród wybranych problemów znajduje się co najmniej jeden potwierdzony problem.
Convert to cause	Zaznacz pole wyboru, aby przekształcić problem(y) objawowy(e) w problem(y) przyczynowy(e).
Close problem	Zaznacz pole wyboru, aby ręcznie zamknąć wybrany(e) problem(y). Pole wyboru zamykania problemu jest dostępne, jeśli opcja *Allow manual close* jest zaznaczona w konfiguracji wyzwalacza dla co najmniej jednego z wybranych problemów. Po kliknięciu przycisku *Update* zostaną zamknięte tylko problemy, które można zamknąć. Jeśli żadnego problemu nie można zamknąć ręcznie, pole wyboru jest wyłączone. Problemy już zamknięte nie zostaną zamknięte ponownie.
Suppress trigger	Kliknij, aby rozpocząć doraźny okres konserwacji dla wyzwalacza. Zostanie otwarty formularz konfiguracji okresu konserwacji. Zostanie w nim wstępnie wybrany jednorazowy okres konserwacji trwający 1 godzinę (konfigurowalny w ustawieniach użytkownika i jego profilu), rozpoczynający się od teraz. Odpowiedni wyzwalacz również zostanie wstępnie wybrany. Nazwa konserwacji zostanie wstępnie uzupełniona w formacie `Ad-hoc: <event name>`.

Potwierdzanie problemu Jeśli użytkownik otrzyma powiadomienie o zdarzeniu problemowym, może przejść do frontend Zabbix, otworzyć okno podręczne aktualizacji problemu dla tego problemu jedną z metod wymienionych poniżej i potwierdzić problem. Podczas potwierdzania może wpisać komentarz, na przykład że pracuje nad jego rozwiązaniem lub dowolną inną informację, którą uzna za stosowną.

W ten sposób, jeśli inny użytkownik systemu zauważy ten sam problem, natychmiast zobaczy, czy został on potwierdzony, oraz dotychczasowe komentarze.

W ten sposób proces rozwiązywania problemów przez więcej niż jednego użytkownika systemu może przebiegać w skoordynowany sposób.

Status potwierdzenia jest również używany podczas definiowania operacji akcji. Można na przykład określić, że powiadomienie zostanie wysłane do menedżera wyższego szczebla tylko wtedy, gdy zdarzenie nie zostanie potwierdzone przez pewien czas.

Aby potwierdzać zdarzenia i dodawać do nich komentarze, użytkownik musi mieć co najmniej uprawnienia do odczytu odpowiednich wyzwalaczy. Aby zmienić istotność problemu lub zamknąć problem, użytkownik musi mieć uprawnienia do odczytu i zapisu odpowiednich wyzwalaczy.

Istnieje **kilka** sposobów uzyskania dostępu do okna podręcznego aktualizacji problemu, które umożliwia potwierdzenie problemu.

- Możesz zaznaczyć problemy w *Monitoring* > *Problems*, a następnie kliknąć *Mass update* pod listą
- Możesz kliknąć *Update* w kolumnie *Update* problemu w:
  - *Dashboards* (widżety *Problems* i *Problems by severity*)

  - *Monitoring > Problems*
  - *Monitoring > Problems > Event details*
- Możesz kliknąć komórkę nierozwiązanego problemu w:
  - *Dashboards* (widżet *Trigger overview*)

Menu podręczne zawiera opcję *Update*, która przeniesie Cię do okna aktualizacji problemu.

Na podstawie informacji o potwierdzeniach można skonfigurować sposób wyświetlania liczby problemów na pulpicie nawigacyjnym lub mapach. W tym celu należy dokonać wyboru w opcji *Wyświetlanie problemów*, dostępnej zarówno w konfiguracji mapy, jak i w widżecie pulpitu nawigacyjnego *Problemy według ważności*. Możliwe jest wyświetlanie całkowitej liczby problemów, liczby niepotwierdzonych problemów oddzielnie od łącznej liczby lub tylko liczby niepotwierdzonych problemów.

Na podstawie informacji o aktualizacji problemu (potwierdzenie itp.) można skonfigurować operacje aktualizacji - wysyłanie wiadomości lub wykonywanie zdalnych poleceń.

Wyciszanie problemów Wyciszanie problemów umożliwia tymczasowe ukrycie problemu, którym można zająć się później. Jest to przydatne do uporządkowania listy problemów i nadania najwyższego priorytetu najpilniejszym kwestiom. Na przykład czasami w weekend może wystąpić problem, który nie jest na tyle pilny, aby zajmować się nim natychmiast, więc można go „uśpić" do poniedziałku rano.

Wyciszanie problemów umożliwia ukrycie *pojedynczego* problemu, w przeciwieństwie do wyciszania za pomocą konserwacji hosta, podczas której ukrywane są wszystkie problemy hosta objętego konserwacją.

Operacje dla działań wyzwalacza zostaną wstrzymane w przypadku wyciszonych problemów w taki sam sposób, jak ma to miejsce podczas konserwacji hosta.

Problem może zostać wyciszony za pomocą okna **aktualizacji problemu**, gdzie wyciszenie jest jedną z opcji aktualizacji problemu, obok komentowania, zmiany ważności, potwierdzania itd.

Problem może również zostać niewyciszony za pomocą tego samego okna aktualizacji problemu.

Wyświetlanie

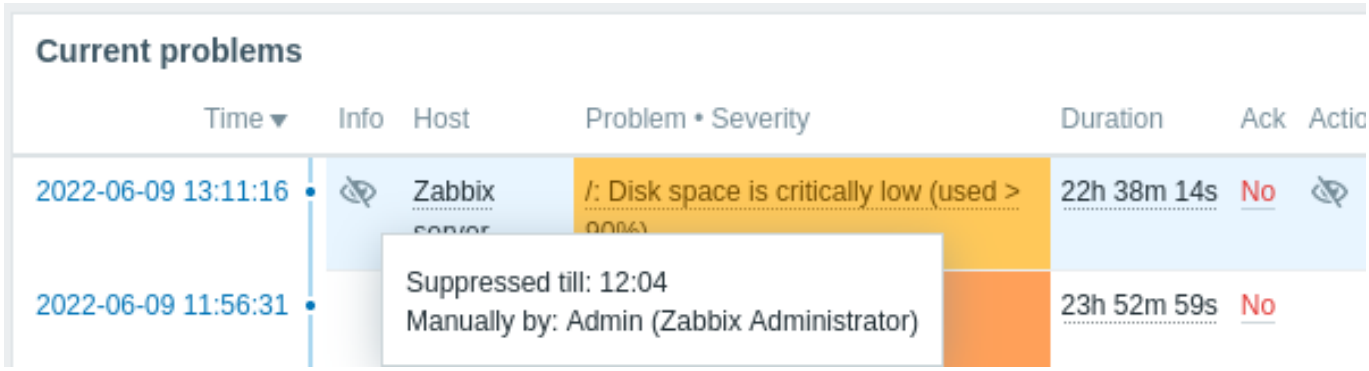
Po wyciszeniu problem jest oznaczany migającą ikoną wyciszenia w kolumnie *Info*, zanim zostanie ukryty.

Jeśli wyciszenie wynika z konserwacji hosta, ikona wyciszenia w kolumnie *Info* jest wyświetlana na pomarańczowo. Jeśli obowiązują zarówno konserwacja, jak i ręczne wyciszenie, ręczne wyciszenie ma pierwszeństwo w kolorze ikony.

Ikona wyciszenia miga, gdy zadanie wyciszenia znajduje się na liście oczekujących. Gdy menedżer zadań wyciszy problem, ikona przestanie migać. Jeśli ikona wyciszenia miga przez długi czas, może to wskazywać na problem z serwerem, na przykład gdy serwer jest niedostępny i menedżer zadań nie może zakończyć zadania. Ta sama logika dotyczy niewyciszania. W krótkim okresie po wysłaniu zadania i zanim serwer je zakończy, ikona niewyciszenia miga.

Wyciszony problem może być zarówno ukryty, jak i widoczny, w zależności od ustawień filtra/widgetu problemów.

Gdy jest widoczny na liście problemów, wyciszony problem jest oznaczany ikoną wyciszenia, a szczegóły wyciszenia są wyświetlane po najechaniu myszą:



Szczegóły wyciszenia są również wyświetlane w oknie podręcznym po umieszczeniu kursora myszy na ikonie wyciszenia w kolumnie *Actions*.

Ręczne zamykanie problemów

Przegląd Chociaż zazwyczaj zdarzenia problemów są rozwiązywane automatycznie, gdy stan wyzwalacza zmienia się z `Problem` na `OK`, mogą wystąpić przypadki, w których trudno jest określić, czy problem został rozwiązany na podstawie wyrażenia wyzwalacza. W takich przypadkach problem należy rozwiązać ręcznie.

Na przykład *syslog* może zgłaszać, że niektóre parametry jądra wymagają dostosowania, aby zapewnić optymalną wydajność. W takim przypadku problem jest zgłaszany administratorom Linuxa, którzy go naprawiają, a następnie zamykają problem ręcznie.

Problemy można zamykać ręcznie tylko dla wyzwalaczy z włączoną opcją *Allow manual close*.

Gdy problem zostanie "zamknięty ręcznie", Zabbix generuje nowe wewnętrzne zadanie dla serwera Zabbix. Następnie proces *task manager* wykonuje to zadanie i generuje zdarzenie OK, zamykając w ten sposób zdarzenie problemu.

Wygenerowane zdarzenie *OK* zawiera pełny zestaw tagów zdarzenia rozwiązanych dla tego zdarzenia (w tym tagi odziedziczone z szablonów, hostów i wyzwalaczy). Tagi te są dostępne w powiadomieniach oraz w makrach takich jak `{EVENT.RECOVERY.TAGS}` i `{EVENT.RECOVERY.TAGSJSON}`.

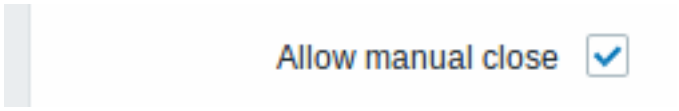
Ręczne zamknięcie problemu nie oznacza, że powiązany wyzwalacz nigdy ponownie nie przejdzie w stan *Problem*. Wyrażenie wyzwalacza jest ponownie oceniane i może skutkować problemem:

- Gdy pojawią się nowe dane dla dowolnej pozycji uwzględnionej w wyrażeniu wyzwalacza (należy pamiętać, że wartości odrzucone przez krok przetwarzania wstępnego ograniczającego częstotliwość nie są uznawane za odebrane i nie spowodują ponownej oceny wyrażenia wyzwalacza);
- Gdy w wyrażeniu są używane funkcje date and time i/lub nodata().

Konfiguracja Do ręcznego zamknięcia problemu wymagane są dwa kroki.

Konfiguracja wyzwalacza

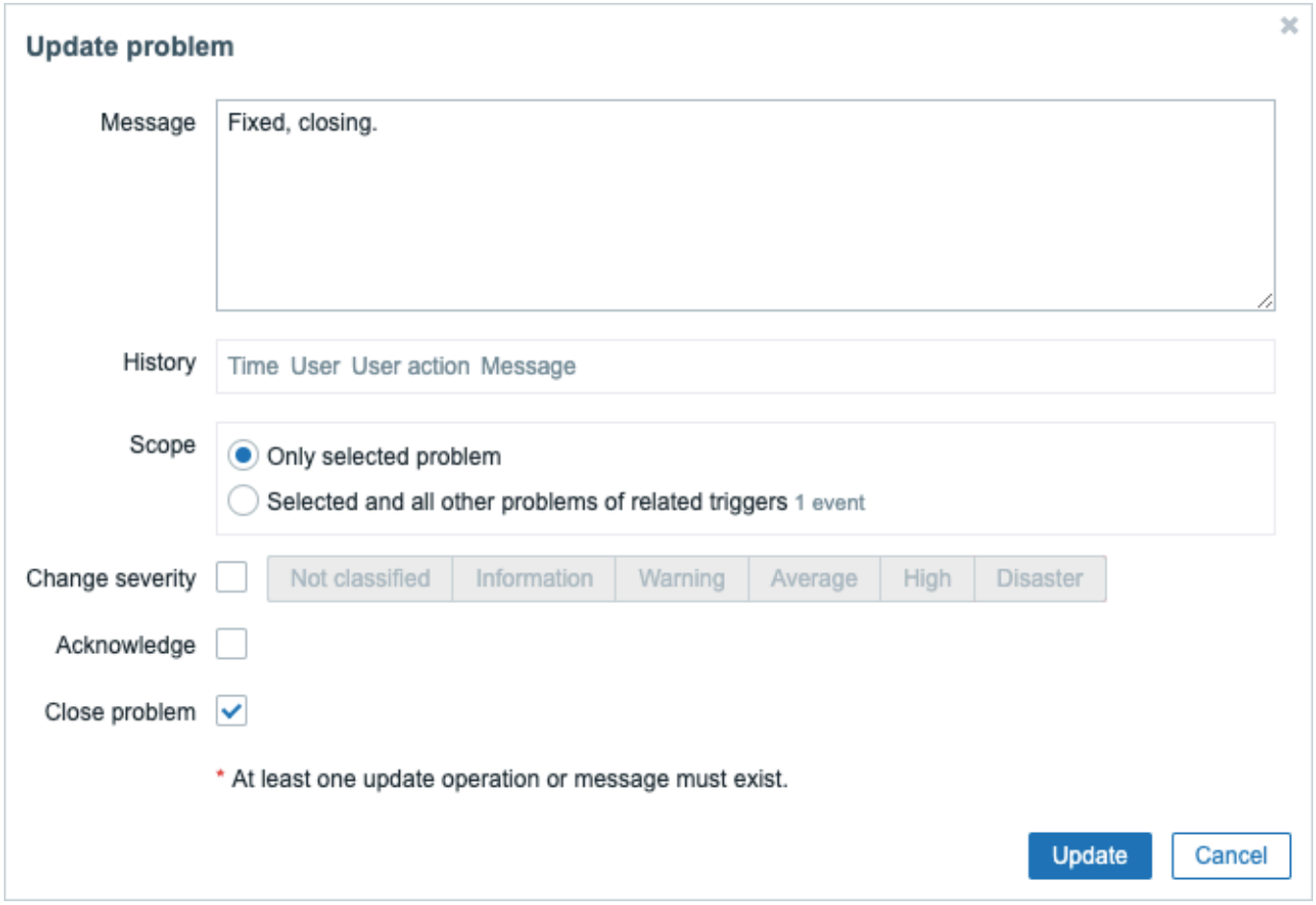
W konfiguracji wyzwalacza włącz opcję *Zezwalaj na ręczne zamknięcie*.



Okno aktualizacji problemu

Jeśli wystąpi problem dla wyzwalacza z flagą *Manual close*, możesz otworzyć okno podręczne aktualizacji problemu dla tego problemu i zamknąć go ręcznie.

Aby zamknąć problem, zaznacz opcję *Close problem* w formularzu i kliknij *Update*.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Żądanie jest przetwarzane przez serwer Zabbix. Zwykle zamknięcie problemu zajmuje kilka sekund. W trakcie tego procesu w *Monitoring* > *Problems* jako status problemu wyświetla się *CLOSING*.

Weryfikacja Można zweryfikować, że problem został zamknięty ręcznie:

- w szczegółach zdarzenia, dostępnych przez *Monitoring* > *Problemy*;
- za pomocą makra `{EVENT.UPDATE.HISTORY}` w wiadomościach powiadomień, które dostarczy te informacje.

Okresy konserwacji

Przegląd

Konserwacja służy do tłumienia problemów w określonych wcześniej okresach.

Okresy konserwacji można definiować dla następujących encji:

- Grup hostów
- Hostów
- Wyzwalaczy
- Nazw zdarzeń
- Znaczników

Jeśli konserwacja jest zdefiniowana dla określonych znaczników lub nazw zdarzeń, zostanie aktywowana tylko dla odpowiednich wyzwalaczy; wszystkie pozostałe wyzwalacze hosta lub grupy hostów nie będą objęte konserwacją.

Istnieją dwa typy konserwacji: *Z gromadzeniem danych* oraz *Bez gromadzenia danych*.

Podczas konserwacji *Z gromadzeniem danych* wyzwalacze są przetwarzane jak zwykle, a zdarzenia są tworzone w razie potrzeby. Jeśli jednak w konfiguracji akcji zaznaczono opcję *Wstrzymaj operacje dla tłumionych problemów*, eskalacje problemów są wstrzymywane dla encji objętych konserwacją. W takim przypadku kroki eskalacji, które mogą obejmować wysyłanie powiadomień lub zdalnych poleceń, będą pomijane przez cały czas trwania okresu konserwacji. Należy pamiętać, że operacje odzyskiwania i aktualizacji nie są tłumione podczas konserwacji — dotyczy to wyłącznie eskalacji. Powiadomienia o odzyskaniu nie są wysyłane, jeśli problem rozpoczął się podczas konserwacji.

Na przykład jeśli kroki eskalacji są zaplanowane na 0, 30 i 60 minut po rozpoczęciu problemu, a po wystąpieniu rzeczywistego problemu rozpoczyna się trwająca pół godziny konserwacja, od 10. do 40. minuty, kroki drugi i trzeci zostaną wykonane pół godziny później, czyli w 60. i 90. minucie (pod warunkiem że problem nadal występuje). Podobnie, jeśli problem wystąpi podczas konserwacji, eskalacja rozpocznie się po zakończeniu konserwacji.

Aby normalnie otrzymywać powiadomienia o problemach podczas konserwacji (bez opóźnienia), należy odznaczyć opcję *Wstrzymaj operacje dla tłumionych problemów* w konfiguracji akcji.

Note:
Jeśli co najmniej jeden host (użyty w wyrażeniu wyzwalacza) nie jest w trybie konserwacji, Zabbix wyśle powiadomienie o problemie.

Serwer Zabbix musi działać podczas konserwacji. Konserwacje są przeliczane co minutę lub natychmiast po ponownym załadowaniu pamięci podręcznej konfiguracji, jeśli wprowadzono zmiany w okresie konserwacji.

Procesy timer sprawdzają, czy stan hosta należy zmienić na tryb konserwacji lub z niego wyłączyć, w 0. sekundzie każdej minuty. Dodatkowo co sekundę proces timer sprawdza, czy należy rozpocząć lub zakończyć konserwacje na podstawie zmian w [okresach konserwacji] po aktualizacji konfiguracji. Dlatego szybkość rozpoczynania i kończenia okresów konserwacji zależy od interwału aktualizacji konfiguracji (domyślnie 10 sekund). Należy pamiętać, że zmiany okresu konserwacji nie obejmują ustawień *Aktywne od/Aktywne do*. Ponadto jeśli encja (grupa hostów/host/wyzwalacz itp.) zostanie dodana do istniejącego aktywnego okresu konserwacji, zmiany zostaną aktywowane przez proces timer dopiero na początku następnej minuty.

Należy pamiętać, że gdy host przechodzi w tryb konserwacji, procesy timer serwera Zabbix odczytają wszystkie otwarte problemy, aby sprawdzić, czy należy je stłumić. Może to mieć wpływ na wydajność, jeśli istnieje wiele otwartych problemów. Serwer Zabbix odczyta również wszystkie otwarte problemy podczas uruchamiania, nawet jeśli w tym momencie nie skonfigurowano żadnych konserwacji.

Należy pamiętać, że serwer Zabbix (lub proxy) zawsze gromadzi dane niezależnie od typu konserwacji (w tym podczas konserwacji *Bez gromadzenia danych*). Jeśli ustawiono opcję *Bez gromadzenia danych*, dane zostaną później zignorowane przez serwer.

Po zakończeniu konserwacji *Bez gromadzenia danych* wyzwalacze używające funkcji `nodata()` nie zostaną uruchomione przed następnym sprawdzeniem w okresie, który sprawdzają.

Jeśli pozycja dziennika zostanie dodana, gdy host jest w trybie konserwacji, a następnie konserwacja się zakończy, zostaną zebrane tylko nowe wpisy pliku dziennika od momentu zakończenia konserwacji.

Jeśli wartość ze znacznikiem czasu zostanie wysłana dla hosta objętego konserwacją *Bez gromadzenia danych* (np. za pomocą Zabbix sender), wartość ta zostanie odrzucona; można jednak wysłać wartość ze znacznikiem czasu przypadającym na zakończony okres konserwacji i zostanie ona zaakceptowana.

Jeśli użytkownik zmieni okres konserwacji lub opcję konserwacji (grupę hostów, hosta, wyzwalacz itp.), zmiany zaczną obowiązywać dopiero po synchronizacji pamięci podręcznej konfiguracji.

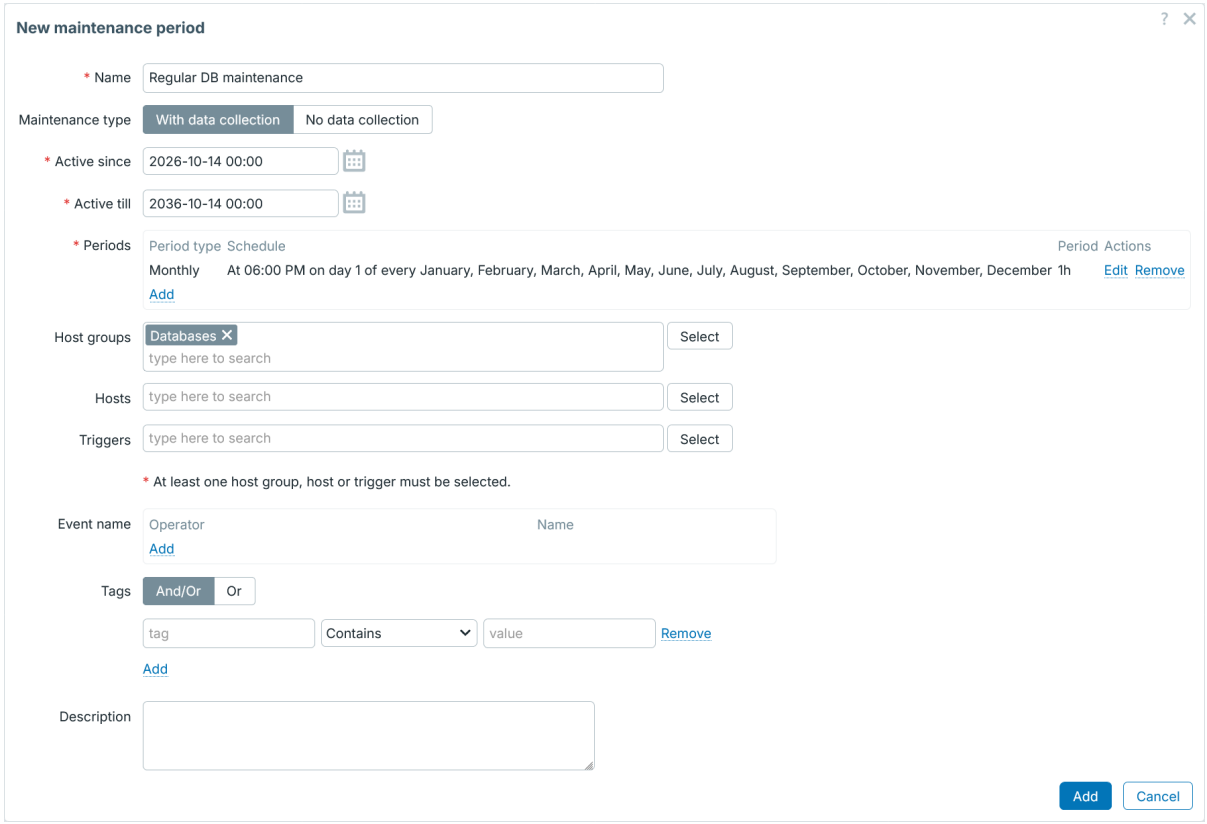
Konfiguracja

Aby natychmiast rozpocząć okres konserwacji:

1. Przejdź do *Monitoring* > *Problems*.
2. Kliknij nazwę problemu.
3. Wybierz opcję konserwacji z menu zdarzenia (host, wyzwalacz, nazwa zdarzenia, tagi).
4. Zmodyfikuj parametry konserwacji w formularzu.

Aby skonfigurować zaplanowany okres konserwacji:

1. Przejdź do *Data collection* > *Maintenance*.
2. Kliknij *Create maintenance period* (lub nazwę istniejącego okresu konserwacji).
3. Wprowadź parametry konserwacji w formularzu.



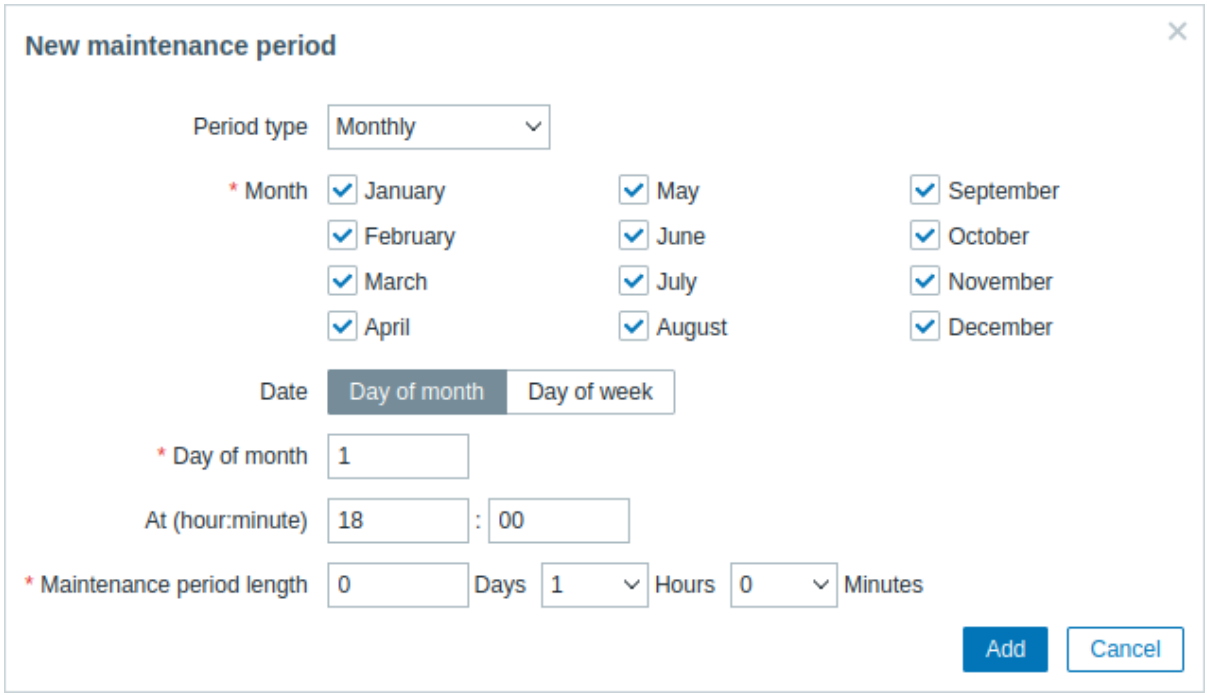
Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Nazwa okresu konserwacji.
Maintenance type	Można ustawić dwa typy konserwacji: **With data collection** - serwer zbiera dane podczas konserwacji, a wyzwalacze są przetwarzane; **No data collection** - dane mogą być nadal zbierane, ale nie są zapisywane w bazie danych podczas konserwacji, a wyzwalacze (w tym funkcja `nodata()`) nie są uruchamiane. Informacje o wpływie każdego typu na raport dostępności zawiera sekcja Wpływ okresów konserwacji.
Active since	Data i godzina, od których wykonywanie okresów konserwacji staje się aktywne. *Uwaga:* Samo ustawienie tej godziny nie aktywuje okresu konserwacji; okresy konserwacji muszą być skonfigurowane w sekcji *Periods* (zobacz poniżej).
Active till	Data i godzina, do których wykonywanie okresów konserwacji pozostaje aktywne.
Periods	Ten blok umożliwia zdefiniowanie dokładnych dni i godzin, w których odbywa się konserwacja. Kliknięcie Add otwiera wyskakujące okno z elastycznym formularzem *Maintenance period*, w którym można zdefiniować harmonogram konserwacji. Szczegółowy opis zawiera sekcja Okresy konserwacji.
Host groups	Wybierz grupy hostów, dla których zostanie aktywowana konserwacja. Konserwacja zostanie aktywowana dla wszystkich hostów z określonej grupy lub grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania wyświetli listę rozwijaną wszystkich dostępnych grup hostów. Określenie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. W związku z tym konserwacja zostanie również aktywowana dla hostów z grup zagnieżdżonych.
Hosts	Wybierz hosty, dla których zostanie aktywowana konserwacja. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania wyświetli listę rozwijaną wszystkich dostępnych hostów.
Triggers	Wybierz wyzwalacze, dla których zostanie aktywowana konserwacja. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania wyświetli listę rozwijaną wszystkich dostępnych wyzwalaczy.

Parameter	Description
Event name	Określ nazwy zdarzeń, dla których zostanie aktywowana konserwacja. Można ustawić kilka warunków. Dla każdego warunku dostępne są dwa operatory: **Contains** - uwzględnia określone nazwy tagów, jeśli wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter) **Does not contain** - wyklucza określone nazwy tagów, jeśli wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter)
Tags	Określ tagi, dla których zostanie aktywowana konserwacja. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Contains** - uwzględnia określone nazwy tagów, jeśli wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu z uwzględnieniem wielkości liter); **Equals** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter). **Does not equal** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter) **Does not contain** - wyklucza określone nazwy tagów, jeśli wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter) Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone operatorem Or; **Or** - wystarczy spełnienie jednego warunku. Tagi można określić tylko wtedy, gdy wybrano typ konserwacji *With data collection*.
Description	Opis okresu konserwacji.

Okresy konserwacji

Okno okresu konserwacji służy do planowania czasu dla konserwacji cyklicznej lub jednorazowej. Formularz jest dynamiczny, a dostępne pola zmieniają się w zależności od wybranego *Typu okresu*.



Typ okresu	Opis
Jednorazowo	Skonfiguruj jednorazowy okres konserwacji: *Data* - data i godzina okresu konserwacji; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna.

Typ okresu	Opis
Dziennie	Skonfiguruj dzienny okres konserwacji: *Co dzień/dni* - częstotliwość konserwacji (1 - *(domyślnie)* codziennie, 2 - co dwa dni itd.); *O (godzina:minuta)* - godzina rozpoczęcia konserwacji w ciągu dnia; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna. Gdy parametr *Co dzień/dni* jest większy niż ”1”, dniem początkowym jest dzień, na który przypada czas *Aktywne od*. Przykłady: - jeśli *Aktywne od* jest ustawione na ”2021-01-01 12:00”, *Co dzień/dni* jest ustawione na ”2”, a *O (godzina:minuta)* jest ustawione na ”23:00”, to pierwszy okres konserwacji rozpocznie się 1 stycznia o 23:00, a drugi okres konserwacji rozpocznie się 3 stycznia o 23:00; - jeśli *Aktywne od* jest ustawione na ”2021-01-01 12:00”, *Co dzień/dni* jest ustawione na ”2”, a *O (godzina:minuta)* jest ustawione na ”01:00”, to pierwszy okres konserwacji rozpocznie się 3 stycznia o 01:00, a drugi okres konserwacji rozpocznie się 5 stycznia o 01:00.
Tygodniowo	Skonfiguruj tygodniowy okres konserwacji: *Co tydzień/tygodnie* - częstotliwość konserwacji (1 - *(domyślnie)* co tydzień, 2 - co dwa tygodnie itd.); *Dzień tygodnia* - w który dzień ma odbywać się konserwacja; *O (godzina:minuta)* - godzina rozpoczęcia konserwacji w ciągu dnia; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna. Gdy parametr *Co tydzień/tygodnie* jest większy niż ”1”, tygodniem początkowym jest tydzień, na który przypada czas *Aktywne od*. Przykłady znajdują się w opisie parametru *Dziennie* powyżej.
Miesięcznie	Skonfiguruj miesięczny okres konserwacji: *Miesiąc* - wybierz wszystkie miesiące, w których jest wykonywana regularna konserwacja; *Data:* ***Dzień miesiąca*** - wybierz tę opcję, jeśli konserwacja ma odbywać się w tym samym dniu każdego miesiąca (na przykład każdego 1. dnia miesiąca), a następnie wybierz wymagany dzień w polu *Dzień miesiąca*, które się pojawi; *Data:* ***Dzień tygodnia*** - wybierz tę opcję, jeśli konserwacja ma odbywać się tylko w określone dni (na przykład w każdy pierwszy poniedziałek miesiąca), a następnie wybierz (z listy rozwijanej) wymagany tydzień miesiąca (pierwszy, drugi, trzeci, czwarty lub ostatni), a potem zaznacz pola wyboru dla dnia/dni konserwacji; *O (godzina:minuta)* - godzina rozpoczęcia konserwacji w ciągu dnia; *Długość okresu konserwacji* - jak długo konserwacja będzie aktywna.

Attention:
Podczas tworzenia okresu konserwacji używana jest strefa czasowa użytkownika, który go tworzy. Jednak podczas planowania cyklicznych okresów konserwacji (*Daily*, *Weekly*, *Monthly*) używana jest strefa czasowa serwera Zabbix. Aby zapewnić przewidywalne działanie cyklicznych okresów konserwacji, wymagane jest używanie wspólnej strefy czasowej we wszystkich częściach Zabbix.

Gdy skończysz, kliknij *Add*, aby dodać okres konserwacji do sekcji *Periods*.

Należy pamiętać, że zmiany czasu letniego (DST) nie wpływają na długość trwania konserwacji. Na przykład załóżmy, że skonfigurowano dwugodzinną konserwację, która zwykle rozpoczyna się o 01:00 i kończy o 03:00:

- jeśli po jednej godzinie konserwacji (o 02:00) nastąpi zmiana czasu letniego i bieżący czas zmieni się z 02:00 na 03:00, konserwacja będzie trwać jeszcze jedną godzinę (do 04:00);
- jeśli po dwóch godzinach konserwacji (o 03:00) nastąpi zmiana czasu letniego i bieżący czas zmieni się z 03:00 na 02:00, konserwacja zostanie zakończona, ponieważ minęły dwie godziny;
- jeśli okres konserwacji rozpoczyna się w godzinie pomijanej podczas zmiany czasu letniego, konserwacja nie zostanie uruchomiona.

Jeśli okres konserwacji jest ustawiony na ”1 day” (rzeczywisty okres konserwacji wynosi 24 godziny, ponieważ Zabbix oblicza dni w godzinach), rozpoczyna się o 00:00 i kończy o 00:00 następnego dnia:

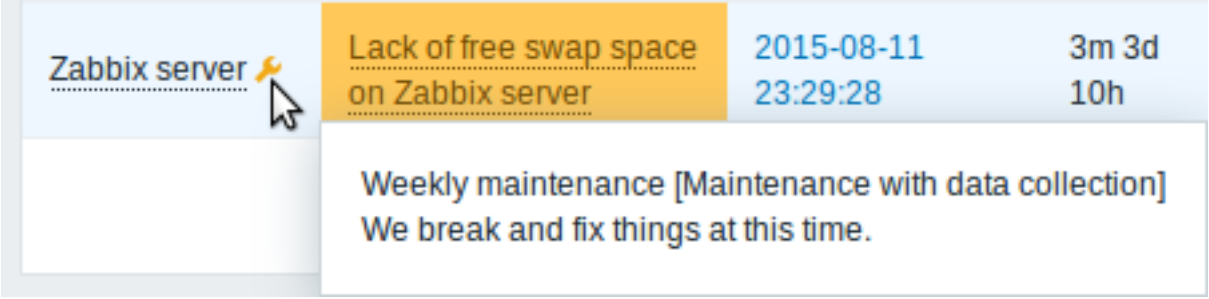
- konserwacja zakończy się o 01:00 następnego dnia, jeśli czas zostanie przesunięty o jedną godzinę do przodu;
- konserwacja zakończy się o 23:00 tego samego dnia, jeśli czas zostanie przesunięty o jedną godzinę do tyłu.

Wyświetlanie

Wyświetlanie hostów w trybie konserwacji

Pomarańczowa ikona klucza  obok nazwy hosta wskazuje, że ten host jest w trybie konserwacji w:

- *Dashboards*
- *Monitoring* > *Problems*
- *Inventory* > *Hosts* > *Host inventory details*
- *Data collection* > *Hosts* (zobacz kolumnę *Status*)



Szczegóły konserwacji są wyświetlane po umieszczeniu wskaźnika myszy nad ikoną.

Dodatkowo hosty w trybie konserwacji mają pomarańczowe tło w *Monitoring* > *Maps*.

Wyświetlanie ukrytych problemów

Zwykle problemy dla hostów w trybie konserwacji są ukrywane, to znaczy nie są wyświetlane we frontendzie.

Ukryte problemy mogą być również niewidoczne, ponieważ zostały ręcznie ukryte.

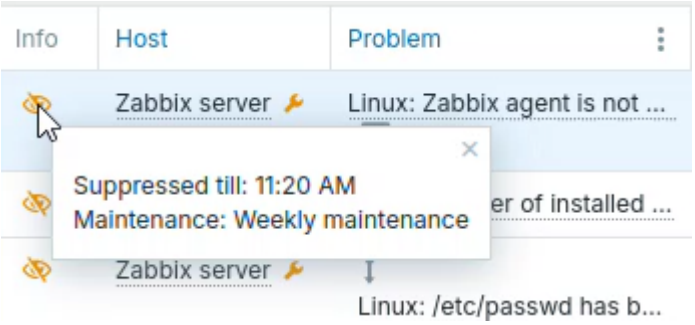
Można wyświetlać ukryte problemy, wybierając opcję *Pokaż ukryte problemy* w następujących miejscach:

- *Pulpity* (w konfiguracji widżetów *Hosty z problemami*, *Problemy*, *Problemy według ważności*, *Przegląd wyzwalaczy*)
- *Monitoring* > *Problemy* (w filtrze)
- *Monitoring* > *Mapy* (w konfiguracji mapy)
- Globalne powiadomienia (w konfiguracji profilu użytkownika)

Gdy ukryte problemy są wyświetlane, widoczna jest następująca ikona: . Jeśli ukrycie wynika z konserwacji hosta, wyświetlane są następujące ikony:

- - problem został ukryty z powodu konserwacji hosta.
- - problem został odblokowany po zakończeniu konserwacji hosta.

Najechanie kursorem na ikonę wyświetla więcej szczegółów. W szczegółach zdarzenia można sprawdzić, czy zdarzenie zostało ukryte z powodu konserwacji, czy ukryte ręcznie. Jeśli mają zastosowanie zarówno konserwacja, jak i ręczne ukrycie, ręczne ukrycie ma pierwszeństwo w kolorze ikony.



Obliczanie kolejek podczas konserwacji

Kolejki wyświetlane we frontendzie Zabbix (Administration > Queue) są obliczane przez serwer Zabbix. Nie obejmują one pozycji w konserwacji *No data collection*, a długość kolejki dla tych pozycji zawsze wynosi zero, nawet jeśli ich wartości są opóźnione. Opóźnione pozycje w konserwacji *With data collection* są nadal uwzględniane w kolejce.

proxy Zabbix nie jest świadomy okresów konserwacji, ponieważ nie ma synchronizacji konfiguracji konserwacji między serwerem Zabbix a proxy. Kontrole wewnętrzne obliczane na proxy Zabbix (na przykład `zabbix[queue,,]` i `zabbix[stats,,,queue,,]`) raportują opóźnione pozycje niezależnie od stanu konserwacji na serwerze Zabbix.

W rezultacie frontend Zabbix i kontrole wewnętrzne na proxy Zabbix mogą raportować różne długości kolejki dla tych samych pozycji w konserwacji *No data collection*.

Tagowanie

Tagi mają wiele zastosowań, a przede wszystkim służą do oznaczania zdarzeń. Gdy encje są oznaczone tagami, każde nowe zdarzenie związane z oznaczoną encją odziedziczy jej tagi. Na przykład:

- w przypadku oznaczonych szablonów - każdy problem hosta (utworzony przez wyzwalacze z szablonu) odziedziczy tagi szablonu.
- w przypadku oznaczonych hostów - każdy problem hosta odziedziczy tagi hosta.
- w przypadku oznaczonych pozycji/scenariuszy web - każdy problem pozycji/scenariusza web odziedziczy tagi pozycji/scenariusza web.
- w przypadku oznaczonych wyzwalaczy - każdy problem utworzony przez wyzwalacz odziedziczy tagi wyzwalacza.

Zdarzenie problemu dziedziczy wszystkie tagi z całego łańcucha encji - szablonów, hostów, pozycji/scenariuszy web, wyzwalaczy. Identyczne kombinacje `tag:value` (po rozwiązaniu makr) są scalane w jedną, co pozwala uniknąć duplikacji.

Zdarzenia odzyskania generowane przez ręczne zamknięcie również zawierają rozwiązane tagi zdarzenia odziedziczone z szablonów, hostów, pozycji/scenariuszy web i wyzwalaczy. Tagi te są dostępne w powiadomieniach oraz za pomocą makr, takich jak `{EVENT.RECOVERY.TAGS}` i `{EVENT.RECOVERY.TAGSJSON}`.

Niestandardowe tagi zdarzeń oferują większą elastyczność. Na przykład:

- korelacja zdarzeń może być skonfigurowana na podstawie tagów zdarzeń.
- warunki akcji mogą być skonfigurowane na podstawie tagów zdarzeń.
- problemy pozycji mogą być grupowane na podstawie tagów zdarzeń.
- tagi problemów mogą być używane do mapowania problemów na usługi.

Encje mogą być oznaczone tym samym nazwą tagu, ale z różnymi wartościami tagu (na przykład `component:memory` i `component:storage`). Podobnie encja może mieć tag bez wartości oraz ten sam tag z wartością (na przykład `database` i `database:postgresql`). Takie tagi nie są uznawane za duplikaty.

Tagi składają się z nazwy tagu i wartości tagu. Podczas oznaczania encji można użyć samej nazwy albo połączyć ją z wartością (na przykład `mysql`, `jira`, `target:mysql`, `service:jira` itd.).

Tagi można definiować dla różnych encji:

- Szablony
- Hosty
- Pozycje
- Scenariusze web
- Wyzwalacze
- Usługi
- Pozycje i wyzwalacze szablonów
- Prototypy hostów, pozycji i wyzwalaczy

Na listach *Szablony*, *Hosty*, *Pozycje*, *Wyzwalacze*, *Scenariusze web* oraz ich prototypów kolumna *Tagi* wyświetla zarówno tagi zwykłe, jak i odziedziczone. Po najechaniu kursorem na tag *odziedziczony* lub kliknięciu go pojawia się podpowiedź JavaScript z tekstem "Inherited tag". Jeśli odziedziczony tag istnieje również jako tag zwykły, wyświetlany jest inny tekst podpowiedzi w zależności od listy, na której się pojawia (na przykład "Inherited and template tag" na liście szablonów lub "Inherited and host tag" na liście hostów). W innych miejscach frontend tagi zwykłe i odziedziczone są wyświetlane razem (duplikaty są usuwane), bez ikon i bez dodatkowego tekstu podpowiedzi.

Note:
Zapoznaj się z oficjalnymi wytycznymi Zabbixa dotyczącymi ogólnych zaleceń dotyczących definiowania tagów, a także ze szczegółowymi wskazówkami dla szablonów, pozycji, wyzwalaczy oraz reguł wykrywania niskiego poziomu.

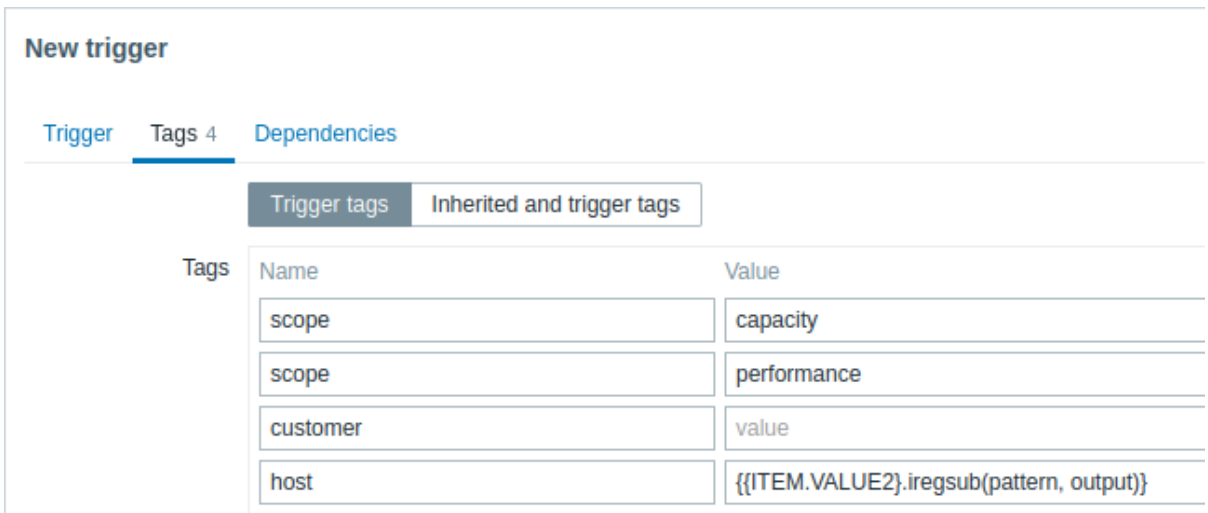
Przypadki użycia Oto kilka typowych przypadków użycia tagowania:

1. Oznaczanie zdarzeń wyzwalacza:
   - Zdefiniuj tag wyzwalacza (na przykład `scope:performance`).
   - Problemy utworzone przez ten wyzwalacz będą miały tag wyzwalacza.
2. Oznaczanie problemów odziedziczonych po szablonie:
   - Zdefiniuj tag szablonu (na przykład `target:mysql`).
   - Problemy utworzone przez wyzwalacze z tego szablonu będą miały tag szablonu.
3. Oznaczanie problemów hosta:
   - Zdefiniuj tag hosta (na przykład `service:jira`).

- Problemy utworzone przez wyzwalacze z tego hosta będą miały tag hosta.

4. Filtrowanie powiązanych pozycji:
   - Zdefiniuj tag pozycji (na przykład `component:cpu`).
   - W *Monitoring* > *Latest data* pozycje można filtrować według tagu `component:cpu`.
5. Używanie informacji wyodrębnionych z wartości pozycji jako wartości tagu:
   - Zdefiniuj tag z makrem jako wartością tagu (na przykład `tag-name:{{ITEM.VALUE<N>}.regsub()}`).
   - W *Monitoring* > *Problems* problemy będą miały wartość tagu rozwiązaną do danych wyodrębnionych z wartości pozycji.
6. Identyfikowanie problemów w pliku dziennika i zamykanie ich osobno:
   - Zdefiniuj tag wyzwalacza dla wyzwalacza pozycji monitorującej logi, który będzie wyodrębniał wartości z wartości pozycji za pomocą makra (na przykład `service:{{ITEM.VALUE<N>}.regsub()}`).
   - W konfiguracji wyzwalacza skonfiguruj korelację zdarzeń:
     - ustaw *PROBLEM event generation mode* na "Multiple";
     - ustaw *OK event closes* na "All problems if tag values match";
     - ustaw tag do dopasowania.
   - Problemy utworzone przez wyzwalacz pozycji logu będą miały tag wyzwalacza i będą zamykane indywidualnie.
7. Filtrowanie powiadomień:
   - Zdefiniuj tagi wyzwalacza (na przykład `scope:security` dla trigger1 i `scope:availability` dla trigger2).
   - Użyj filtrowania tagów w warunkach akcji, aby otrzymywać powiadomienia tylko o zdarzeniach zgodnych z danymi tagów.
8. Identyfikowanie problemów w powiadomieniach:
   - Zdefiniuj tagi wyzwalacza.
   - Użyj makra {EVENT.TAGS} w powiadomieniu o problemie.
   - Powiadomienie o problemie będzie zawierać tagi wyzwalacza, co ułatwi identyfikację, do której aplikacji/usługi należy powiadomienie.
9. Uproszczenie zadań konfiguracyjnych przez użycie tagów szablonu:
   - Zdefiniuj tag wyzwalacza szablonu.
   - Wyzwalacze utworzone z tego wyzwalacza szablonu będą miały jego tag.
10. Tworzenie wyzwalaczy z tagami z niskopoziomowego wykrywania (LLD):
    - Zdefiniuj tag prototypu wyzwalacza z makrem LLD w nazwie lub wartości tagu (na przykład `scope:{#FSNAME}`).
    - Wyzwalacze utworzone z prototypu wyzwalacza będą miały jego tag.
11. Dopasowywanie usług za pomocą tagów usług:
    - Zdefiniuj tagi usług.
    - Skonfiguruj akcje usług dla usług z pasującymi tagami.
    - Dodatkowo użyj tagów usług, aby powiązać usługę z SLA na potrzeby obliczeń SLA.
12. Powiązywanie usług z problemami za pomocą tagów problemów usług:
    - Zdefiniuj tag problemu w konfiguracji usługi (na przykład `target:mysql`).
    - Problemy z pasującym tagiem zostaną automatycznie skorelowane z usługą, a stan usługi zmieni się zgodnie ze skonfigurowanymi regułami obliczania stanu usługi.
13. Wstrzymywanie problemów, gdy host jest w trybie konserwacji:
    - Zdefiniuj tagi w konfiguracji okresu konserwacji.
    - Problemy z zdefiniowanymi tagami zostaną wstrzymane.
14. Przyznawanie dostępu grupom użytkowników:
    - Zdefiniuj tagi w konfiguracji grupy użytkowników.
    - Użytkownicy w grupie użytkowników będą mogli wyświetlać tylko problemy z zdefiniowanymi tagami.

Konfiguracja Tagi można zdefiniować na dedykowanej karcie, na przykład w konfiguracji wyzwalacza:



Obsługa makr Wbudowane i makra użytkownika w tagach są rozwiązywane w momencie wystąpienia zdarzenia. Do czasu wystąpienia zdarzenia te makra będą wyświetlane w frontend Zabbix jako nierozwiązane.

Makra wykrywania niskiego poziomu są rozwiązywane podczas procesu wykrywania.

W nazwach i wartościach tagów wyzwalacza można używać następujących makr:

- {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} oraz {HOST.ID} - makra wbudowane
- {INVENTORY.*} - makra wbudowane (do odwoływania się do wartości inwentarza hosta z jednego lub kilku hostów w wyrażeniu wyzwalacza)
- Makra użytkownika oraz makra użytkownika z kontekstem (kontekst może zawierać makra wykrywania niskiego poziomu)
- Makra wykrywania niskiego poziomu (tylko w tagach prototypu wyzwalacza)

W nazwach i wartościach tagów szablonu, hosta oraz pozycji/scenariusza web można używać następujących makr:

- {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} oraz {HOST.ID} - makra wbudowane
- {INVENTORY.*} - makra wbudowane
- Makra użytkownika
- Makra wykrywania niskiego poziomu (tylko w tagach prototypu hosta i pozycji)

W powiadomieniach opartych na wyzwalaczach można używać następujących makr:

- {EVENT.TAGS} i {EVENT.RECOVERY.TAGS} - makra wbudowane (te makra zostaną rozwinięte do listy tagów zdarzenia lub tagów zdarzenia odzyskania, rozdzielonych przecinkami)
- {EVENT.TAGSJSON} i {EVENT.RECOVERY.TAGSJSON} - makra wbudowane (te makra zostaną rozwinięte do tablicy JSON zawierającej obiekty tagów zdarzenia objects lub obiekty tagów zdarzenia odzyskania)

Wyodrębnianie podciągu w tagach wyzwalacza

Wyodrębnianie podciągu jest obsługiwane podczas wypełniania nazwy tagu lub wartości tagu przy użyciu funkcji makra function. Funkcja stosuje wyrażenie regularne do wartości uzyskanej przez obsługiwane makro. Na przykład:

```
{{ITEM.VALUE}.regsub(pattern, output)}
{{ITEM.VALUE}.iregsub(pattern, output)}

{{#LLDMACRO}.regsub(pattern, output)}
{{#LLDMACRO}.iregsub(pattern, output)}
```

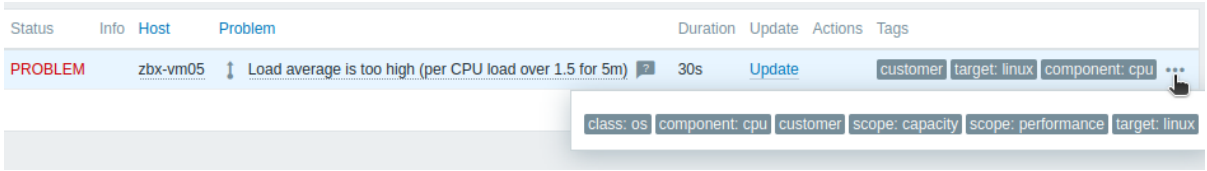
Jeśli nazwa tagu lub jego wartość przekroczy 255 znaków po rozwinięciu makra, zostanie skrócona do 255 znaków.

Zobacz też: Używanie funkcji makr w makrach niskopoziomowego wykrywania do tagowania zdarzeń.

Wyświetlanie tagów zdarzeń Tagi, jeśli są zdefiniowane, można wyświetlać wraz z nowymi zdarzeniami w:

- *Monitoring* > *Problems*
- *Monitoring* > *Problems* > *Event details*
- *Dashboards* > *Problems* widget

Kolejność i liczba wyświetlanych tagów są określane przez opcje filtrowania *Tag display priority* i *Show tags* w *Monitoring* > *Problems* lub w widżecie pulpitu *Problems*. Należy pamiętać, że można wyświetlić maksymalnie trzy tagi; jeśli tagów jest więcej, po najechaniu kursorem na trzy kropki wszystkie tagi zostaną pokazane w oknie podręcznym.



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.

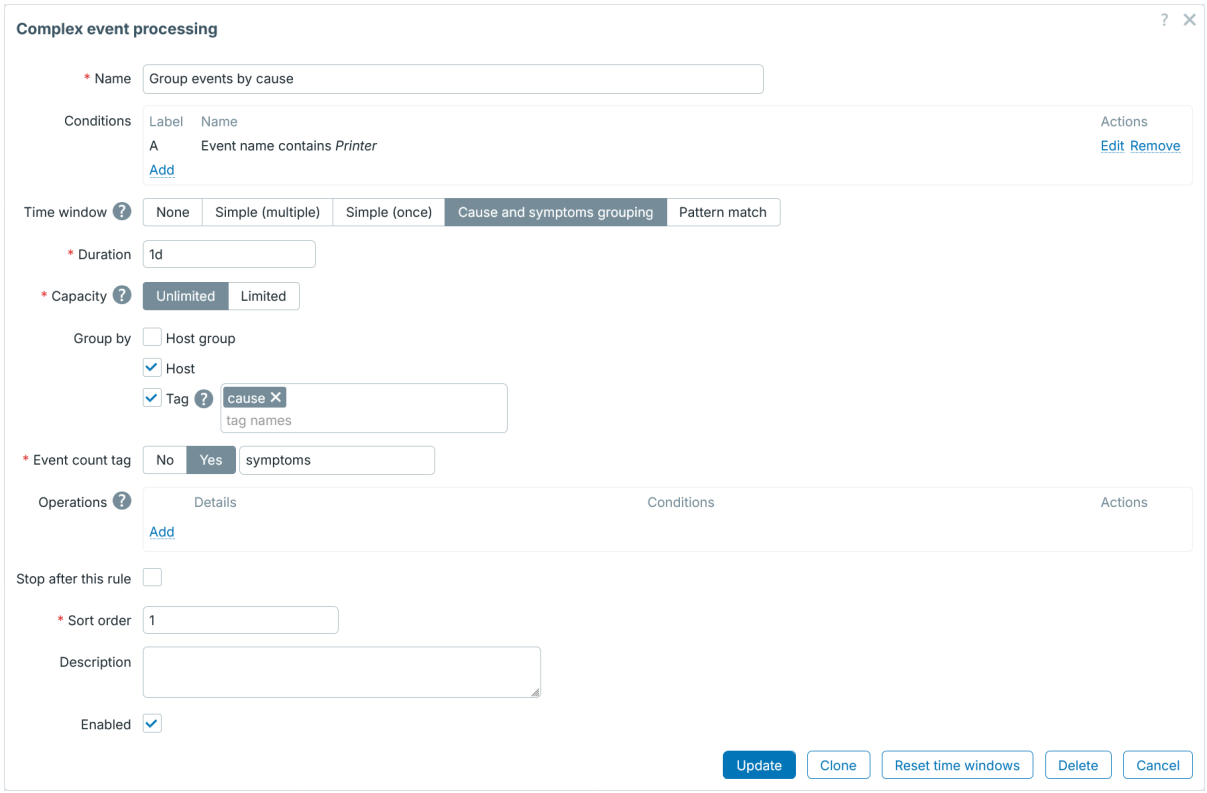
Event stream analysis is defined by CEP rules. A CEP rule consists of a time window (optional), conditions and operations. Operations are tied to an execution point.

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:

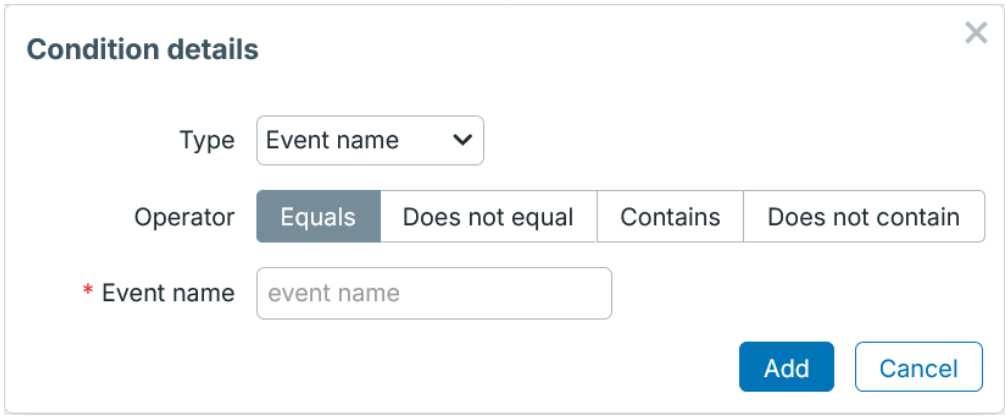
1. Go to *Data collection* > *Event processing*.
2. Click *Create complex event processing* in the upper-right corner of the screen.
3. Enter parameters of the rule in the form.



Parameter	Description
Name	Specify a unique CEP rule name.
Conditions	Specify conditions for associating events with this CEP rule.

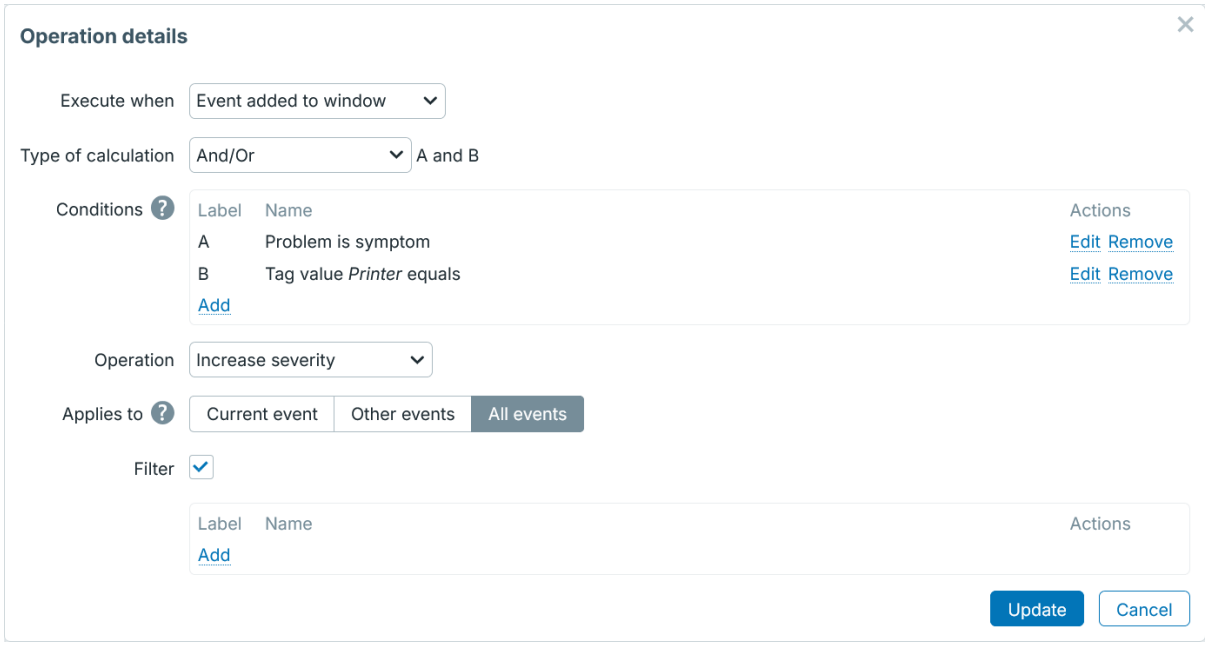
Parameter	Description
Time window	Select the time window type: **None** - process single events with no time window (default) **Simple (multiple)** - process events in a sliding time window (event can be added to multiple windows of this type) **Simple (once)** - process events in a sliding time window (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.
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.

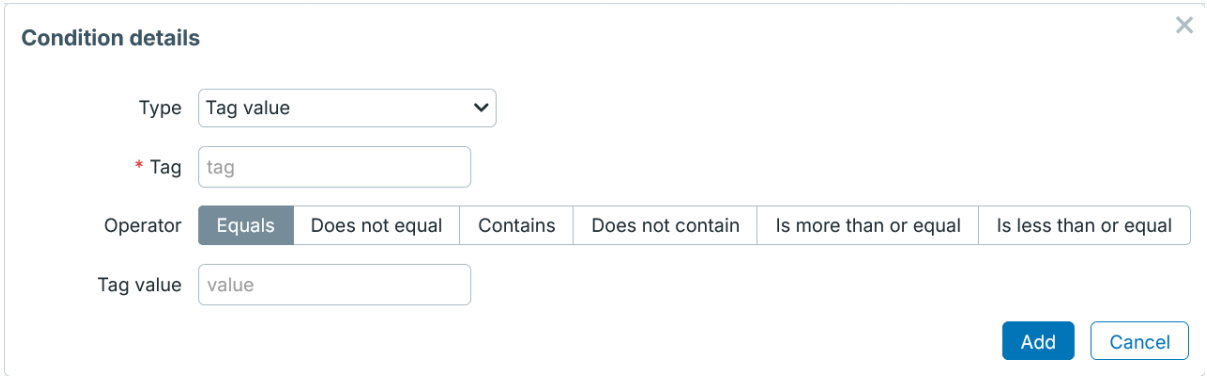
CEP rule operation



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.

Parameter	Description
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.

Korelacja zdarzeń

Przegląd

Korelacja zdarzeń umożliwia powiązanie zdarzeń problemowych z ich rozwiązaniem w bardzo precyzyjny i elastyczny sposób.

Korelację zdarzeń można definiować:

- Na poziomie wyzwalacza - jeden wyzwalacz może służyć do powiązania oddzielnych problemów z ich rozwiązaniem;
- Globalnie - problemy można powiązać z ich rozwiązaniem pochodzącym z innego wyzwalacza/metody odpytywania za pomocą globalnych reguł korelacji.

Korelacja zdarzeń oparta na wyzwalaczach

Przegląd

Korelacja zdarzeń oparta na wyzwalaczach umożliwia korelowanie oddzielnych problemów zgłaszanych przez jeden wyzwalacz.

Zasadniczo zdarzenie OK może zamknąć wszystkie zdarzenia problemowe utworzone przez jeden wyzwalacz, jednak w niektórych przypadkach potrzebne jest bardziej szczegółowe podejście. Na przykład podczas monitorowania plików dziennika możesz chcieć wykrywać określone problemy w pliku dziennika i zamykać je pojedynczo, a nie wszystkie naraz.

Dotyczy to wyzwalaczy, dla których parametr *PROBLEM event generation mode* jest ustawiony na *Multiple*. Takie wyzwalacze są zwykle używane do monitorowania dzienników, przetwarzania trapów itp.

W Zabbixie można powiązać zdarzenia problemowe na podstawie znaczników. Znaczniki służą do wyodrębniania wartości i tworzenia identyfikatorów zdarzeń problemowych. Dzięki temu problemy można również zamykać pojedynczo na podstawie zgodnego znacznika.

Innymi słowy, ten sam wyzwalacz może tworzyć oddzielne zdarzenia identyfikowane przez znacznik zdarzenia. Dzięki temu zdarzenia problemowe można identyfikować pojedynczo i zamykać oddzielnie na podstawie identyfikacji za pomocą znacznika zdarzenia.

Jak to działa

W monitorowaniu logów możesz napotkać wiersze podobne do tych:

```
Wiersz 1: Usługa 1 zatrzymana
Wiersz 2: Usługa 2 zatrzymana
Wiersz 3: Usługa 1 została uruchomiona ponownie
Wiersz 4: Usługa 2 została uruchomiona ponownie
```

Ideą korelacji zdarzeń jest możliwość dopasowania zdarzenia problemu z Wiersza 1 do rozwiązania z Wiersza 3 oraz zdarzenia problemu z Wiersza 2 do rozwiązania z Wiersza 4, a następnie zamykania tych problemów jeden po drugim:

```
Wiersz 1: Usługa 1 zatrzymana
Wiersz 3: Usługa 1 została uruchomiona ponownie #problem z Wiersza 1 zamknięty

Wiersz 2: Usługa 2 zatrzymana
Wiersz 4: Usługa 2 została uruchomiona ponownie #problem z Wiersza 2 zamknięty
```

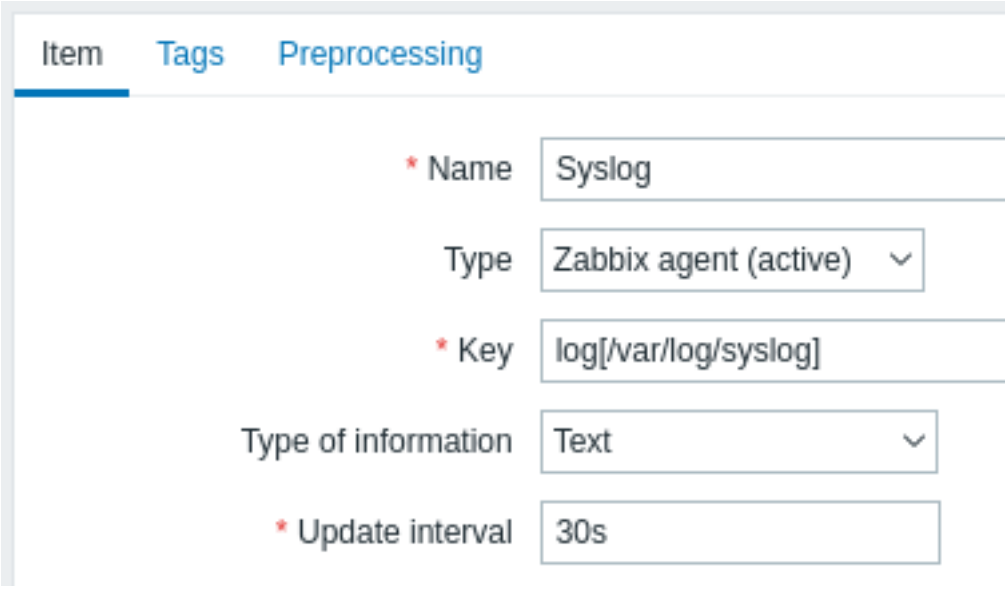
Aby to zrobić, musisz oznaczyć te powiązane zdarzenia na przykład jako ”Usługa 1” i ”Usługa 2”. Można to zrobić, stosując wyrażenie regularne do wiersza logu, aby wyodrębnić wartość znacznika. Następnie, gdy zdarzenia są tworzone, otrzymują odpowiednio znaczniki ”Usługa 1” i ”Usługa 2”, a problem może zostać dopasowany do rozwiązania.

Konfiguracja

Pozycja

Na początek możesz chcieć skonfigurować pozycję, która monitoruje plik dziennika, na przykład:

```
log[/var/log/syslog]
```



Po skonfigurowaniu pozycji poczekaj minutę, aż zmiany konfiguracji zostaną uwzględnione, a następnie przejdź do Latest data, aby upewnić się, że pozycja zaczęła zbierać dane.

Trigger

Przy działającej pozycji musisz skonfigurować wyzwalacz. Ważne jest, aby zdecydować, które wpisy w pliku dziennika są warte uwagi. Na przykład następujące wyrażenie wyzwalacza będzie wyszukiwać ciąg taki jak 'Stopping', aby sygnalizować potencjalne problemy:

```
find(/My host/log[/var/log/syslog],,"regexp","Stopping")=1
```

Attention:
Aby upewnić się, że każda linia zawierająca ciąg "Stopping" jest uznawana za problem, ustaw także *Tryb generowania zdarzeń Problem* w konfiguracji wyzwalacza na 'Multiple'.

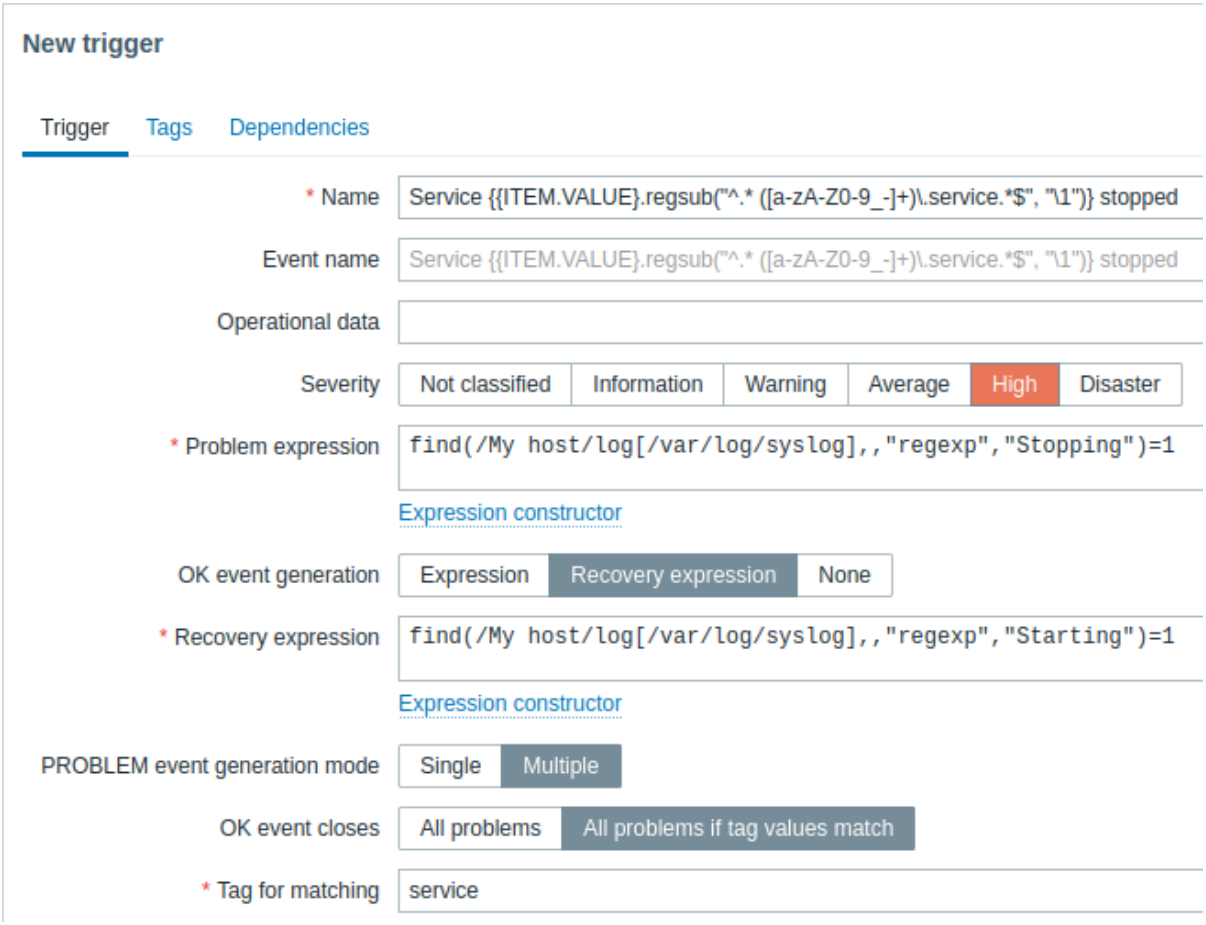
Następnie zdefiniuj wyrażenie odzyskiwania. Poniższe wyrażenie odzyskiwania rozwiąże wszystkie problemy, jeśli zostanie znaleziona linia dziennika zawierająca ciąg "Starting":

```
find(/My host/log[/var/log/syslog],,"regexp","Starting")=1
```

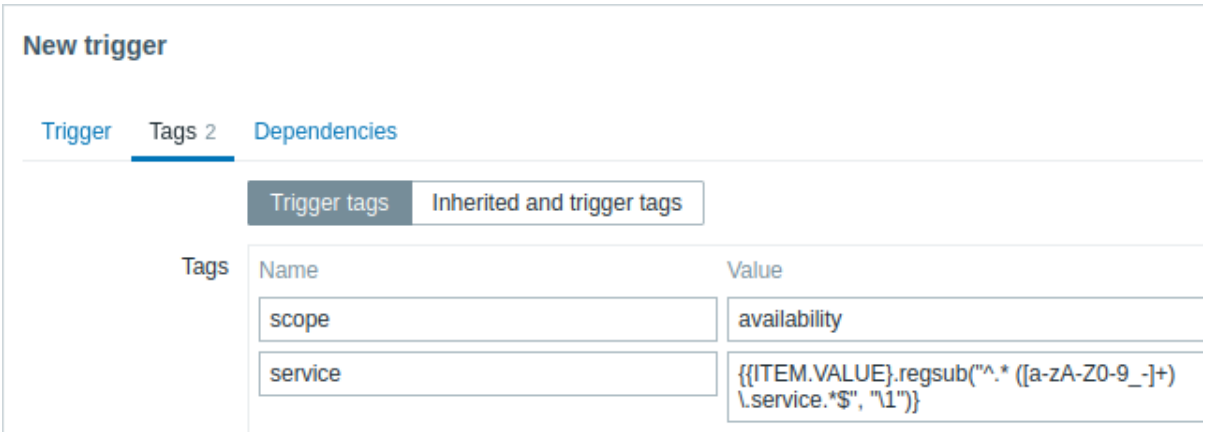
Ponieważ nie chcemy tego, ważne jest, aby w jakiś sposób upewnić się, że odpowiadające im problemy źródłowe są zamykane, a nie tylko wszystkie problemy. W tym może pomóc tagowanie.

Problemy i rozwiązania można dopasować, określając tag w konfiguracji wyzwalacza. Należy wprowadzić następujące ustawienia:

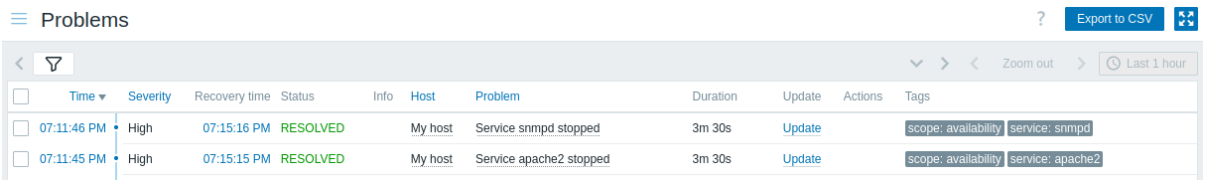
- *Tryb generowania zdarzeń Problem*: Multiple
- *Zdarzenie OK zamyka*: Wszystkie problemy, jeśli wartości tagów są zgodne
- Wprowadź nazwę tagu do dopasowywania zdarzeń



- skonfiguruj tagi, aby wyodrębniać wartości tagów z linii dziennika



Jeśli konfiguracja przebiegnie pomyślnie, w *Monitoring* > *Problemy* będzie można zobaczyć zdarzenia problemów oznaczone według aplikacji i dopasowane do ich rozwiązania.



Warning:
Ponieważ możliwa jest nieprawidłowa konfiguracja, a podobne tagi zdarzeń mogą być tworzone dla **niepowiązanych** problemów, przejrzyj proszę przypadki opisane poniżej!

- Zindeksowane makra zawsze odnoszą się do pola *Expression* w konfiguracji wyzwalacza, a nie do pola *Recovery expression*. Na przykład w zdarzeniu odzyskiwania **{ITEM.VALUE1}** zostanie rozwiązane do najnowszej wartości pierwszej pozycji **w wyrażeniu problemu** w momencie odzyskiwania. Jeśli wyrażenie odzyskiwania opiera się na innej pozycji, a wartość pozycji z wyrażenia problemu zmieni się do czasu odzyskiwania, zdarzenia otrzymają różne tagi i nie zostaną skorelowane.

- Gdy dwie aplikacje zapisują komunikaty o błędach i odzyskaniu do tego samego pliku dziennika, użytkownik może zdecydować się na użycie dwóch tagów *service* w tym samym wyzwalaczu z różnymi wartościami tagów, stosując oddzielne wyrażenia regularne w wartościach tagów, aby wyodrębnić nazwy, powiedzmy, usługi A i usługi B z makra {ITEM.VALUE} (np. gdy formaty komunikatów się różnią). Jednak może to nie zadziałać zgodnie z planem, jeśli nie będzie dopasowania do wyrażeń regularnych. Niedopasowane wyrażenia regularne spowodują puste wartości tagów, a pojedyncza pusta wartość tagu zarówno w zdarzeniach problemu, jak i OK wystarczy do ich skorelowania. W rezultacie komunikat odzyskiwania z usługi A może przypadkowo zamknąć komunikat o błędzie z usługi B.
- Rzeczywiste tagi i wartości tagów stają się widoczne dopiero po wyzwoleniu wyzwalacza. Jeśli użyte wyrażenie regularne jest nieprawidłowe, zostanie ono bez komunikatu zastąpione ciągiem *UNKNOWN*. Jeśli początkowe zdarzenie problemu z wartością tagu *UNKNOWN* zostanie pominięte, mogą pojawić się kolejne zdarzenia OK z tą samą wartością tagu *UNKNOWN*, które mogą zamknąć zdarzenia problemów, których nie powinny były zamknąć.
- Jeśli użytkownik używa makra {ITEM.VALUE} bez funkcji makr jako wartości tagu, obowiązuje ograniczenie do 255 znaków. Gdy komunikaty dziennika są długie, a pierwsze 255 znaków nie zawiera informacji rozróżniających, może to również skutkować podobnymi tagami zdarzeń dla niepowiązanych problemów.

Globalna korelacja zdarzeń

Przegląd

Globalna korelacja zdarzeń umożliwia objęcie korelacją wszystkich metryk monitorowanych przez Zabbix.

Możliwe jest korelowanie zdarzeń utworzonych przez całkowicie różne wyzwalacze i stosowanie do nich wszystkich tych samych operacji. Tworząc inteligentne reguły korelacji, można uniknąć tysięcy powtarzających się powiadomień i skupić się na przyczynach źródłowych problemu!

Globalna korelacja zdarzeń to zaawansowany mechanizm, który pozwala uniezależnić się od logiki problemów i ich rozwiązywania opartej na jednym wyzwalaczu. Do tej pory pojedyncze zdarzenie problemu było tworzone przez jeden wyzwalacz i byliśmy zależni od tego samego wyzwalacza przy rozwiązywaniu problemu. Nie mogliśmy rozwiązać problemu utworzonego przez jeden wyzwalacz za pomocą innego wyzwalacza. Jednak dzięki korelacji zdarzeń opartej na tagach zdarzeń jest to możliwe.

Na przykład wyzwalacz dziennika może zgłaszać problemy aplikacji, podczas gdy wyzwalacz odpytywania może zgłaszać, że aplikacja działa. Korzystając z tagów zdarzeń, można oznaczyć wyzwalacz dziennika tagiem *status:down*, a wyzwalacz odpytywania tagiem *status:up*. Następnie w globalnej regule korelacji można powiązać te wyzwalacze i przypisać do tej korelacji odpowiednią operację, taką jak zamknięcie starych zdarzeń.

W innym zastosowaniu globalna korelacja może identyfikować podobne wyzwalacze i stosować do nich tę samą operację. Co by było, gdyby można było otrzymywać tylko jeden raport o problemie dla każdego problemu z portem sieciowym? Nie ma potrzeby zgłaszania ich wszystkich. Globalna korelacja zdarzeń również to umożliwia.

Globalną korelację zdarzeń konfiguruje się w **regułach korelacji**. Reguła korelacji określa, w jaki sposób nowe zdarzenia problemów są dopasowywane do istniejących zdarzeń problemów oraz co należy zrobić w przypadku dopasowania (zamknąć nowe zdarzenie, zamknąć dopasowane stare zdarzenia, generując odpowiednie zdarzenia OK). Jeśli problem zostanie zamknięty przez globalną korelację, informacja o tym jest wyświetlana w kolumnie *Info* w sekcji *Monitoring* > *Problems*.

Konfigurowanie globalnych reguł korelacji jest dostępne tylko dla użytkowników na poziomie Super Admin.

> **Attention:**
> Korelację zdarzeń należy konfigurować bardzo ostrożnie, ponieważ może ona negatywnie wpłynąć na wydajność przetwarzania zdarzeń lub, jeśli zostanie skonfigurowana nieprawidłowo, zamknąć więcej zdarzeń, niż zamierzano (w najgorszym przypadku mogą zostać zamknięte nawet wszystkie zdarzenia problemów).

Aby **bezpiecznie** skonfigurować globalne reguły korelacji, należy przestrzegać następujących ważnych wskazówek:

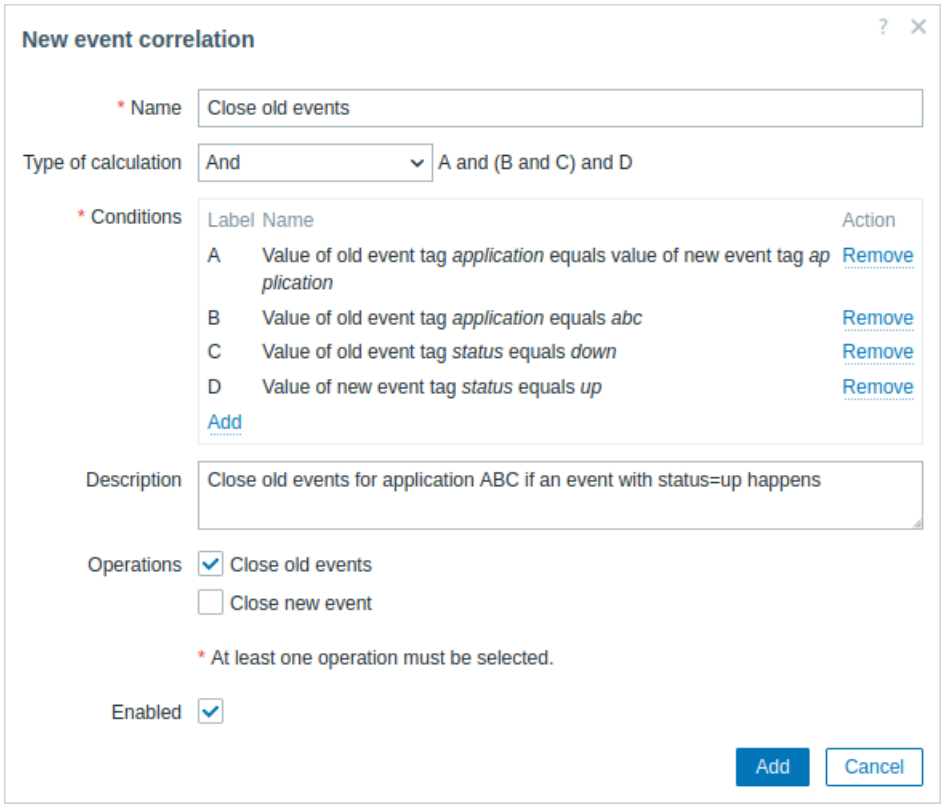
- Ogranicz zakres korelacji. Zawsze ustawiaj unikatowy tag dla nowego zdarzenia, które jest dopasowywane do starych zdarzeń, i używaj warunku korelacji *New event tag name*.
- Dodaj jawny warunek dotyczący starego zdarzenia podczas korzystania z opcji **Close old events**. Zawsze dodawaj co najmniej jeden warunek *Old event* (na przykład *Old event tag name*, *Old event tag value* lub *Event tag pair*) podczas wybierania opcji **Close old events**; w przeciwnym razie reguła może dopasować i zamknąć niezwiązane istniejące zdarzenia problemów (w najgorszym przypadku wszystkie problemy). Preferuj opcję *Event tag pair* do dopasowywania wartości uzyskiwanych w czasie działania (host:port, identyfikator sesji itp.) i, jeśli to możliwe, dodatkowo zawężaj dopasowanie według hosta lub grupy hostów.
- Unikaj używania typowych nazw tagów, które mogą być później używane przez różne konfiguracje korelacji.
- Ogranicz liczbę reguł korelacji do tych, których rzeczywiście potrzebujesz.

Zobacz także: znane problemy.

Konfiguracja

Aby skonfigurować globalnie reguły korelacji zdarzeń:

1. Przejdź do *Data collection* > *Event correlation*.
2. Kliknij *Create event correlation* po prawej stronie (lub nazwę korelacji, aby edytować istniejącą regułę).
3. Wprowadź parametry reguły korelacji w formularzu.

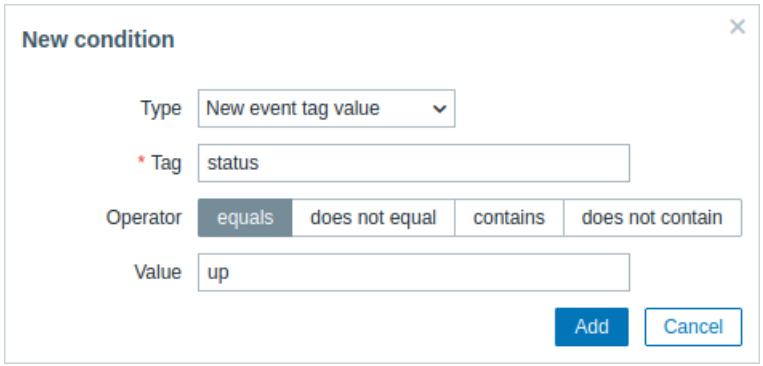


Wszystkie obowiązkowe pola są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Unikalna nazwa reguły korelacji.
Type of calculation	Dostępne są następujące opcje obliczania warunków: **And** - wszystkie warunki muszą być spełnione **Or** - wystarczy, że spełniony jest jeden warunek **And/Or** - AND dla różnych typów warunków i OR dla tego samego typu warunku **Custom expression** - zdefiniowana przez użytkownika formuła obliczeniowa do oceny warunków akcji. Musi zawierać wszystkie warunki (reprezentowane przez wielkie litery A, B, C, ...) i może zawierać spacje, tabulatory, nawiasy ( ), **and** (z uwzględnieniem wielkości liter), **or** (z uwzględnieniem wielkości liter), **not** (z uwzględnieniem wielkości liter).
Conditions	Lista warunków. Szczegóły konfiguracji warunku znajdują się poniżej.
Description	Opis reguły korelacji.
Operations	Zaznacz pole wyboru operacji, która ma zostać wykonana, gdy zdarzenie zostanie skorelowane. Dostępne są następujące operacje: **Close old events** - zamyka stare zdarzenia, gdy wystąpi nowe zdarzenie. Zawsze dodawaj warunek oparty na starym zdarzeniu podczas używania operacji *Close old events*, w przeciwnym razie wszystkie istniejące problemy mogą zostać zamknięte. **Close new event** - zamyka nowe zdarzenie, gdy wystąpi. **Ostrzeżenie!** Nie pozostawiaj pustych warunków starego/nowego zdarzenia podczas używania *Close old events*/*Close new event*. Jeśli wybierzesz operację *Close old events* bez dodania warunku pasującego do starego zdarzenia, Zabbix może dopasować wszystkie istniejące stare zdarzenia i zamknąć je. Zawsze dodawaj jawny warunek starego zdarzenia (na przykład *Old event tag name* lub *Event tag pair*) podczas używania *Close old events*. Na przykład reguła, która używa tylko *New event condition* i operacji *Close old events*, dopasuje wszystkie stare zdarzenia spełniające (brakujące) kryteria starego zdarzenia - w praktyce zamykając stare problemy.

Parameter	Description
Enabled	Jeśli zaznaczysz to pole wyboru, reguła korelacji zostanie włączona.

Aby skonfigurować szczegóły nowego warunku, kliknij Add w bloku Conditions. Otworzy się okno podręczne, w którym można edytować szczegóły warunku.



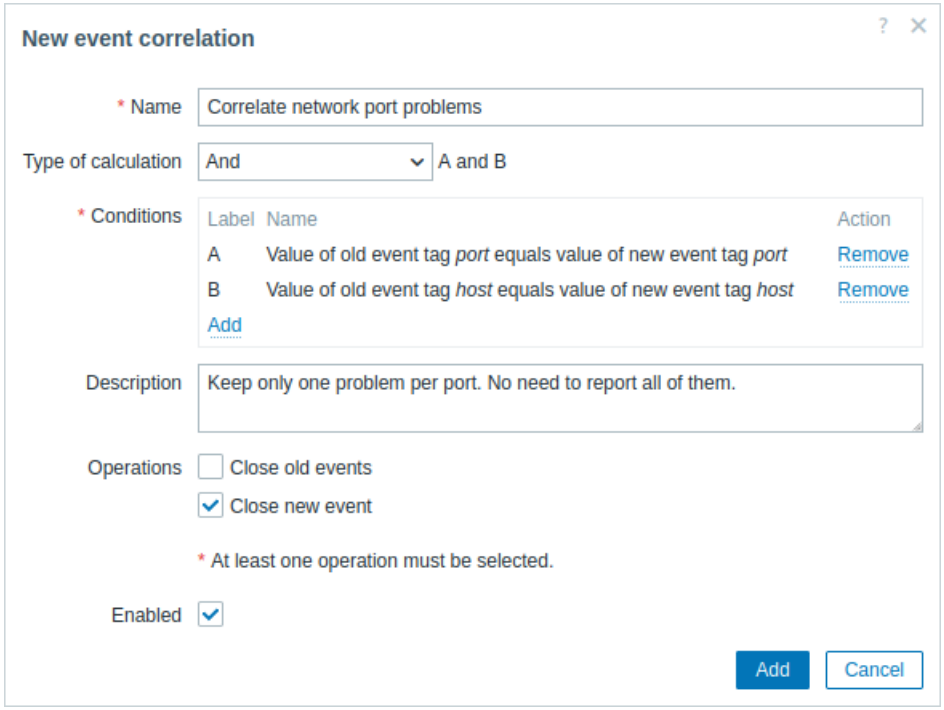
Parameter	Description
New condition	Wybierz warunek do korelowania zdarzeń. *Uwaga*, że jeśli nie zostanie określony warunek starego zdarzenia, wszystkie stare zdarzenia mogą zostać dopasowane i zamknięte. Podobnie, jeśli nie zostanie określony warunek nowego zdarzenia, wszystkie nowe zdarzenia mogą zostać dopasowane i zamknięte. Dostępne są następujące warunki: **Old event tag name** - określ nazwę tagu starego zdarzenia do dopasowania. **New event tag name** - określ nazwę tagu nowego zdarzenia do dopasowania. **New event host group** - określ grupę hostów nowego zdarzenia do dopasowania. **Event tag pair** - określ nazwę tagu nowego zdarzenia i nazwę tagu starego zdarzenia do dopasowania. W tym przypadku dopasowanie nastąpi, jeśli **wartości** tagów w obu zdarzeniach będą zgodne. **Nazwy** tagów nie muszą być zgodne. Ta opcja jest przydatna do dopasowywania wartości w czasie działania, które mogą nie być znane w momencie konfiguracji (zobacz także Przykład). **Old event tag value** - określ nazwę tagu starego zdarzenia i wartość do dopasowania, używając następujących operatorów: *equals* - ma wartość tagu starego zdarzenia *does not equal* - nie ma wartości tagu starego zdarzenia *contains* - zawiera ciąg znaków w wartości tagu starego zdarzenia *does not contain* - nie zawiera ciągu znaków w wartości tagu starego zdarzenia **New event tag value** - określ nazwę tagu nowego zdarzenia i wartość do dopasowania, używając następujących operatorów: *equals* - ma wartość tagu nowego zdarzenia *does not equal* - nie ma wartości tagu nowego zdarzenia *contains* - zawiera ciąg znaków w wartości tagu nowego zdarzenia *does not contain* - nie zawiera ciągu znaków w wartości tagu nowego zdarzenia

Warning:
Ponieważ możliwa jest nieprawidłowa konfiguracja, a podobne tagi zdarzeń mogą być tworzone dla problemów **niepowiązanych**, przejrzyj przypadki opisane poniżej!

- Rzeczywiste tagi i wartości tagów stają się widoczne dopiero po wyzwoleniu wyzwalacza. Jeśli użyte wyrażenie regularne jest nieprawidłowe, zostaje ono bez komunikatu zastąpione ciągiem *UNKNOWN*. Jeśli początkowe zdarzenie problemu z wartością tagu *UNKNOWN* zostanie pominięte, mogą pojawić się kolejne zdarzenia OK z tą samą wartością tagu *UNKNOWN*, które mogą zamknąć zdarzenia problemów, których nie powinny były zamknąć.
- Jeśli użytkownik używa makra {ITEM.VALUE} bez funkcji makr jako wartości tagu, obowiązuje ograniczenie do 255 znaków. Gdy komunikaty dziennika są długie, a pierwsze 255 znaków nie jest jednoznaczne, może to również skutkować podobnymi tagami zdarzeń dla niepowiązanych problemów.

Przykład

Zatrzymaj powtarzające się zdarzenia problemów z tego samego portu sieciowego.



Ta globalna reguła korelacji skoreluje problemy, jeśli na wyzwalaczu istnieją wartości tagów *host* i *port* oraz są one takie same w zdarzeniu pierwotnym i nowym.

Operacja zamknie nowe zdarzenia problemów na tym samym porcie sieciowym, pozostawiając otwarte tylko pierwotne zdarzenie problemu.

Odkrywanie

Przegląd Wykrywanie zapewnia automatyczne sposoby dodawania hostów i konfigurowania monitorowania bez ręcznej konfiguracji. Zabbix obsługuje wiele metod wykrywania, aby dostosować się do różnych scenariuszy.

Metody wykrywania

- Wykrywanie sieci — automatycznie wykrywaj hosty i usługi dostępne w Twojej sieci.
- Autorejestracja aktywnego agenta — umożliwia aktywnym agentom automatyczną rejestrację w serwerze.
- Wykrywanie niskiego poziomu — automatycznie twórz pozycje, wyzwalacze, wykresy i hosty dla wykrytych obiektów, takich jak systemy plików lub interfejsy sieciowe, przy użyciu prototypów i zagnieżdżonych reguł wykrywania.

Odkrywanie sieci

Przegląd

Zabbix oferuje funkcję automatycznego wykrywania sieci, która jest skuteczna i bardzo elastyczna.

Po prawidłowym skonfigurowaniu wykrywania sieci możesz:

- Przyspieszyć wdrożenie Zabbix.
- Uprościć administrację.
- Używać Zabbix w szybko zmieniających się środowiskach bez nadmiernych nakładów administracyjnych.

Wykrywanie sieci w Zabbix opiera się na następujących informacjach:

- Zakresy adresów IP
- Dostępność usług zewnętrznych (FTP, SSH, WEB, POP3, IMAP, TCP itp.)
- Informacje otrzymane od agent Zabbix (obsługiwany jest tylko tryb nieszyfrowany)
- Informacje otrzymane od agent SNMP

Nie zapewnia ono wykrywania topologii sieci.

Wykrywanie sieci zasadniczo składa się z dwóch faz: wykrywania i działań.

Discovery

Zabbix okresowo skanuje zakresy IP zdefiniowane w regułach wykrywania sieci. Częstotliwość sprawdzania można skonfigurować osobno dla każdej reguły.

Każda reguła ma zestaw zdefiniowanych sprawdzeń usług, które mają zostać wykonane dla zakresu IP.

Reguły wykrywania są przetwarzane przez menedżera wykrywania. Menedżer wykrywania tworzy zadanie dla każdej reguły wraz z listą zadań (sprawdzeń sieciowych). Sprawdzenia sieciowe są wykonywane równolegle przez dostępne procesy robocze wykrywania (liczba jest konfigurowalna w frontend dla każdej reguły). Wyjątkiem są sprawdzenia SNMPv3, które są przetwarzane przez pojedynczy proces roboczy. Tylko sprawdzenia z tym samym IP i portem są planowane sekwencyjnie, ponieważ niektóre urządzenia nie akceptują równoległych połączeń na tym samym porcie.

Rozmiar kolejki sprawdzeń sieciowych jest ograniczony do 2000000 lub około 4 GB pamięci. Jeśli kolejka się zapełni, reguła wykrywania zostanie pominięta, a w logu zostanie wyświetlony komunikat ostrzegawczy. Możesz użyć wewnętrznej pozycji `zabbix[discovery_queue]` do monitorowania liczby sprawdzeń wykrywania w kolejce.

Sprawdzenia wykrywania są przetwarzane niezależnie od pozostałych sprawdzeń. Jeśli któreś ze sprawdzeń nie wykryje usługi (lub zakończy się niepowodzeniem), pozostałe sprawdzenia nadal będą przetwarzane.

Note:
Jeśli reguła wykrywania zostanie zmieniona podczas wykonywania, bieżące wykonanie wykrywania zostanie przerwane.

Każde sprawdzenie usługi i hosta (IP) wykonywane przez moduł wykrywania sieci generuje zdarzenie wykrywania.

Event	Check of service result
Service Discovered	Usługa jest 'up' po tym, jak była 'down', lub gdy została wykryta po raz pierwszy.
Service Up	Usługa jest 'up', po tym, jak była już 'up'.
Service Lost	Usługa jest 'down' po tym, jak była 'up'.
Service Down	Usługa jest 'down', po tym, jak była już 'down'.
Host Discovered	Co najmniej jedna usługa hosta jest 'up' po tym, jak wszystkie usługi tego hosta były 'down', albo zostaje wykryta usługa należąca do niezarejestrowanego hosta.
Host Up	Co najmniej jedna usługa hosta jest 'up', po tym jak co najmniej jedna usługa była już 'up'.
Host Lost	Wszystkie usługi hosta są 'down' po tym, jak co najmniej jedna była 'up'.
Host Down	Wszystkie usługi hosta są 'down', po tym jak były już 'down'.

Akcje

Zdarzenia wykrywania mogą stanowić podstawę odpowiednich akcji, takich jak:

- Wysyłanie powiadomień
- Dodawanie/usuwanie hostów
- Włączanie/wyłączanie hostów
- Dodawanie hostów do grupy
- Usuwanie hostów z grupy
- Dodawanie tagów do hosta
- Usuwanie tagów z hosta
- Podłączanie szablonu do hostów/odłączanie szablonu od hostów
- Wykonywanie zdalnych skryptów

Akcje te można skonfigurować z uwzględnieniem typu urządzenia, adresu IP, statusu, czasu działania/przestoju itd. Pełne informacje na temat konfigurowania akcji dla zdarzeń opartych na wykrywaniu sieci można znaleźć na stronach operacji i warunków akcji.

Ponieważ akcje wykrywania sieci są oparte na zdarzeniach, będą uruchamiane zarówno wtedy, gdy wykryty host jest online, jak i wtedy, gdy jest offline. Zdecydowanie zaleca się dodanie warunku akcji *Discovery status: up*, aby uniknąć uruchamiania takich akcji jak *Add host* przy zdarzeniach *Service Lost*/*Service Down*. W przeciwnym razie, jeśli wykryty host zostanie usunięty ręcznie, nadal będzie generował zdarzenia *Service Lost*/*Service Down* i zostanie odtworzony podczas następnego cyklu wykrywania.

Note:
Podłączanie szablonów do wykrytego hosta zakończy się zbiorczym niepowodzeniem, jeśli którykolwiek z szablonów możliwych do podłączenia zawiera unikalną encję (np. klucz pozycji), która jest taka sama jak unikalna encja (np. klucz pozycji) już istniejąca na hoście lub w innym z szablonów możliwych do podłączenia.

Tworzenie hosta

Host jest dodawany, jeśli wybrano operację *Add host*. Host jest również dodawany, nawet jeśli operacja *Add host* nie została określona, jeśli wybierzesz operacje skutkujące działaniami na hoście. Takimi operacjami są:

- Enable host.
- Disable host.
- Add host to a host group.
- Link template to a host.

Utworzone hosty są dodawane do grupy *Discovered hosts* (domyślnie, konfigurowalne w *Administration* > *General* > *Other*). Jeśli chcesz, aby hosty były dodawane do innej grupy, dodaj operację *Remove from host groups* (określając "Discovered hosts") oraz operację *Add to host groups* (określając inną grupę hostów), ponieważ host musi należeć do grupy hostów.

Adres IP wykrytego urządzenia, wraz ze źródłem wykrywania (serwer Zabbix, proxy Zabbix lub grupa proxy) oraz typem interfejsu, jest używany jako kryterium wyszukiwania hosta w systemie. Jeśli host o tym samym adresie IP, typie interfejsu i źródle wykrywania już istnieje, ten host będzie celem wykonywania operacji. Jeśli źródło wykrywania jest inne, wykryty obiekt jest traktowany jako inny host i może zostać utworzony nowy host.

Jeśli adres IP wykrytego hosta zostanie zmieniony lub interfejs zostanie usunięty, przy następnym wykryciu zostanie utworzony nowy host.

Nadawanie nazw hostom

Podczas dodawania hostów nazwa hosta jest wynikiem odwrotnego wyszukiwania DNS lub adresem IP, jeśli odwrotne wyszukiwanie się nie powiedzie. Wyszukiwanie jest wykonywane z serwera Zabbix lub proxy Zabbix, w zależności od tego, który z nich wykonuje wykrywanie. Jeśli wyszukiwanie nie powiedzie się na proxy, nie jest ponawiane na serwerze. Jeśli host o takiej nazwie już istnieje, do nazwy kolejnego hosta zostanie dodane **_2**, następnie **_3** itd.

Możliwe jest również zastąpienie wyszukiwania DNS/IP i użycie zamiast tego wartości pozycji jako nazwy hosta, na przykład:

- Można wykryć wiele serwerów z uruchomionym agent Zabbix przy użyciu pozycji agenta Zabbix do wykrywania i automatycznie przypisać im odpowiednie nazwy na podstawie wartości tekstowej zwróconej przez tę pozycję
- Można wykryć wiele urządzeń sieciowych SNMP przy użyciu pozycji agenta SNMP do wykrywania i automatycznie przypisać im odpowiednie nazwy na podstawie wartości tekstowej zwróconej przez tę pozycję

Jeśli nazwa hosta została ustawiona przy użyciu wartości pozycji, nie jest aktualizowana podczas kolejnych kontroli wykrywania. Jeśli nie można ustawić nazwy hosta przy użyciu wartości pozycji, używana jest wartość domyślna (nazwa DNS).

Jeśli host z wykrytym adresem IP już istnieje, a źródło wykrywania (serwer Zabbix, proxy lub grupa proxy) nie uległo zmianie, nowy host nie jest tworzony. Jeśli źródło wykrywania jest inne, wykryta jednostka jest traktowana jako odrębna i może zostać utworzony nowy host. Jednak jeśli akcja wykrywania zawiera operacje (podłączenie szablonu, dodanie do grupy hostów itp.), są one wykonywane na istniejącym hoście dopasowanym według adresu IP, typu interfejsu i źródła wykrywania.

Usuwanie hostów

Hosty wykryte przez regułę wykrywania sieci są usuwane automatycznie z *Monitoring* > *Discovery*, jeśli wykryta jednostka nie znajduje się już w zakresie adresów IP tej reguły. Hosty są usuwane natychmiast.

Tworzenie interfejsów podczas dodawania hostów

Gdy hosty są dodawane w wyniku wykrywania sieci, tworzone są dla nich interfejsy zgodnie z następującymi zasadami:

- Wykryte usługi - na przykład, jeśli sprawdzenie SNMP zakończyło się powodzeniem, zostanie utworzony interfejs SNMP.
- Jeśli host odpowiedział zarówno na żądania Zabbix agent, jak i SNMP, zostaną utworzone oba typy interfejsów.
- Jeśli kryteria unikalności opierają się na danych zwróconych przez Zabbix agent lub SNMP, pierwszy znaleziony interfejs dla hosta zostanie utworzony jako domyślny. Pozostałe adresy IP zostaną zignorowane, chyba że domyślny interfejs przestanie działać. W przypadku, gdy domyślny interfejs jest niedostępny, jako zamiennik domyślnego interfejsu zostanie użyty najmniejszy z pozostałych adresów IP. Warunki akcji (takie jak Host IP) nie wpływają na dodawanie interfejsów. *Uwaga* - będzie to działać, jeśli wszystkie interfejsy zostaną wykryte przez tę samą regułę wykrywania. Jeśli inna reguła wykrywania wykryje inny interfejs tego samego hosta, zostanie dodany dodatkowy host.
- Jeśli host odpowiedział wyłącznie na sprawdzenia agent, zostanie utworzony tylko z interfejsem agent. Jeśli później zacznie odpowiadać na SNMP, zostaną dodane dodatkowe interfejsy SNMP.
- Jeśli początkowo utworzono 3 oddzielne hosty, wykryte według kryterium unikalności "IP", a następnie reguła wykrywania zostanie zmodyfikowana tak, że hosty A, B i C mają identyczny wynik kryterium unikalności, B i C zostaną utworzone jako dodatkowe interfejsy dla A, pierwszego hosta. Poszczególne hosty B i C pozostają bez zmian. W *Monitoring* > *Discovery* dodane interfejsy będą wyświetlane w kolumnie "Discovered device", czarną czcionką i z wcięciem, natomiast kolumna "Monitored host" będzie wyświetlać tylko A, pierwszy utworzony host. "Uptime/Downtime" nie jest mierzone dla adresów IP, które są uznawane za dodatkowe interfejsy.

Zmiana ustawienia proxy

Hosty wykryte przez różne proxy nie zawsze są traktowane jako różne hosty. Wykrywanie i sprawdzanie unikalności zależą od struktury grupy proxy: gdy proxy uruchamia regułę wykrywania i tworzy hosta, ten host jest dodawany do nadrzędnej grupy proxy tego proxy, a nie przypisywany do samego proxy. Gdy Zabbix ocenia unikalność adresów IP podczas wykrywania, sprawdza hosty monitorowane przez nadrzędną grupę proxy. Hosty monitorowane przez poszczególne proxy w tej grupie (w tym proxy, które uruchomiło wykrywanie) są pomijane podczas sprawdzania unikalności, co może prowadzić do duplikatów hostów, jeśli wiele proxy monitoruje nakładające się podsieci.

Chociaż takie działanie umożliwia wykrywanie w nakładających się zakresach IP używanych przez różne podsieci, zmiana proxy przypisanego do już monitorowanej podsieci jest bardziej złożona, ponieważ zmiany proxy muszą być zastosowane spójnie zarówno do wykrytych hostów, jak i do członkostwa nadrzędnej grupy proxy, aby uniknąć duplikatów.

Na przykład, kroki zastąpienia proxy w regule wykrywania:

1. Wyłącz regułę wykrywania.
2. Zsynchronizuj konfigurację proxy.
3. Zastąp proxy w regule wykrywania.
4. Zastąp proxy dla wszystkich hostów wykrytych przez tę regułę (upewnij się, że hosty w nadrzędnej grupie proxy oraz wszelkie hosty monitorowane przez poszczególne proxy w tej grupie są zaktualizowane, aby uniknąć duplikatów).
5. Włącz regułę wykrywania.

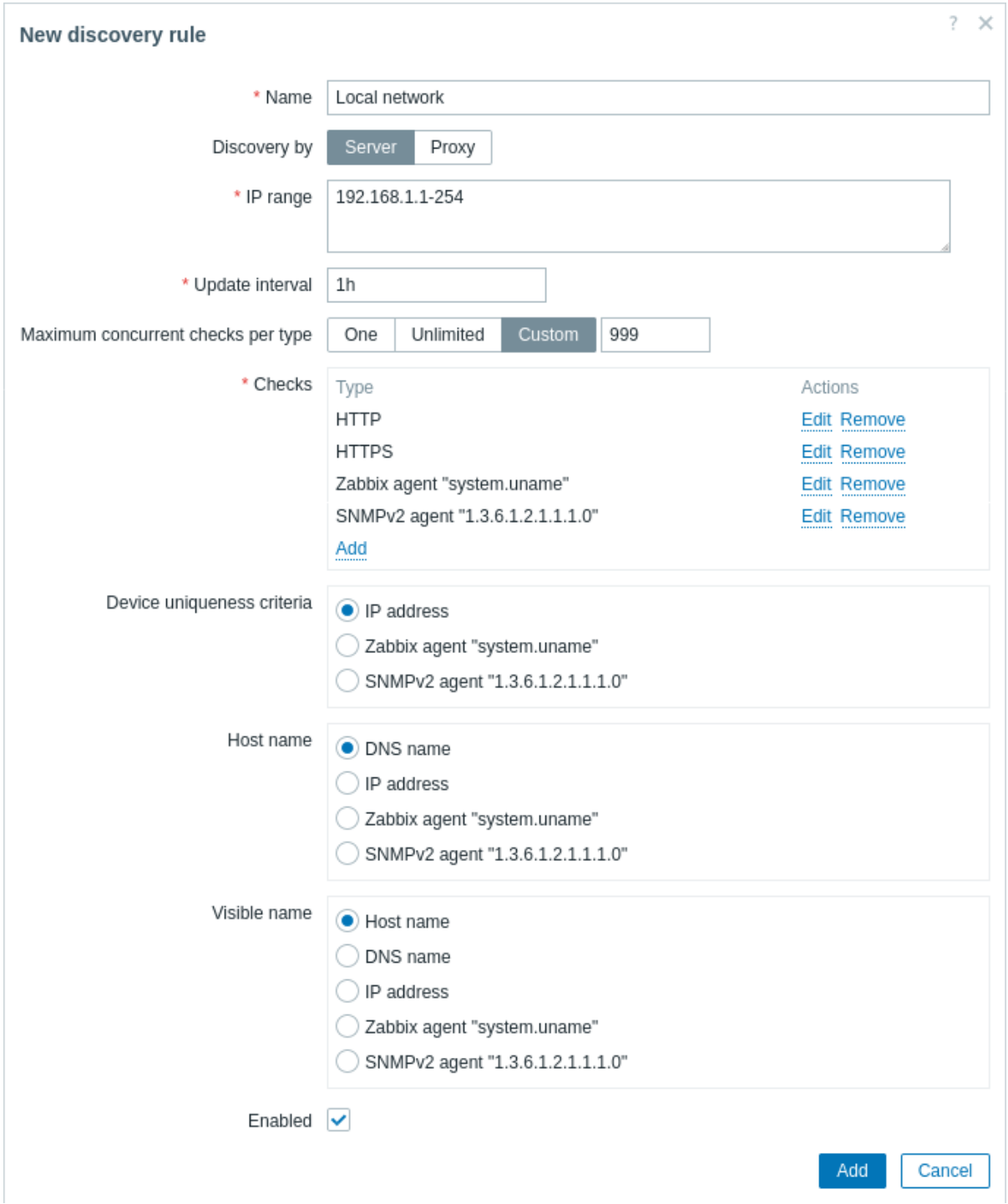
Konfigurowanie reguły wykrywania sieci

Przegląd

Aby skonfigurować regułę wykrywania sieci używaną przez Zabbix do wykrywania hostów i usług:

1. Przejdź do *Data collection* > *Discovery*.
2. Kliknij *Create discovery rule* (lub nazwę reguły, aby edytować istniejącą).
3. Edytuj atrybuty reguły wykrywania.

Atrybuty reguły



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Unikalna nazwa reguły. Na przykład „Local network”.
Discovery by	Wykrywanie jest wykonywane przez: **Server** - przez serwer Zabbix **Proxy** - przez proxy Zabbix (wybrane w polu nazwy proxy)
IP range	Zakres adresów IP do wykrywania. Może mieć następujące formaty: Pojedynczy adres IP: 192.168.1.33 Zakres adresów IP: 192.168.1-10.1-255. Zakres jest ograniczony przez łączną liczbę objętych adresów (mniej niż 64K). Maska IP: 192.168.4.0/24 obsługiwane maski IP: /16 - /30 dla adresów IPv4 /112 - /128 dla adresów IPv6 Lista: 192.168.1.1-255, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24 To pole obsługuje spacje, tabulatory i wiele wierszy.

Parameter	Description
Update interval	Ten parametr określa, jak często Zabbix będzie wykonywać regułę. Interwał jest mierzony po zakończeniu wykonania poprzedniej instancji wykrywania, więc nie dochodzi do nakładania się. Obsługiwane są sufiksy czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga* jeśli używane jest makro użytkownika i jego wartość zostanie zmieniona (np. 1w → 1h), następne sprawdzenie zostanie wykonane zgodnie z poprzednią wartością (w przykładzie: znacznie później).
Maximum concurrent checks per type	Ustaw maksymalną liczbę wątków wykrywania (workerów) na sprawdzenie usługi dla równoległego przetwarzania sprawdzeń wykrywania: **One** - jeden wątek **Unlimited** - nieograniczona liczba wątków (ale nie większa niż w parametrze StartDiscoverers) **Custom** - ustaw własną liczbę wątków (0-999). Asynchroniczne sprawdzenia usług SNMPv1 i SNMPv2 są przetwarzane równolegle przez dostępnych workerów. Wszystkie reguły wykrywania z asynchronicznymi sprawdzeniami usług SNMPv3 są przetwarzane przez jednego workera ze względu na specyfikę implementacji libsnmp; dlatego zwiększenie liczby workerów nie zwiększy szybkości wykrywania.
Checks	Zabbix użyje tej listy sprawdzeń do wykrywania. Kliknij Add, aby skonfigurować nowe sprawdzenie w oknie podręcznym. Obsługiwane sprawdzenia: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Telnet, Zabbix agent, SNMPv1 agent, SNMPv2 agent, SNMPv3 agent, ICMP ping. Wykrywanie oparte na protokole używa funkcjonalności **net.tcp.service[]** do testowania każdego host, z wyjątkiem SNMP, które odpyta o OID SNMP. Zabbix agent jest testowany przez odczyt pozycja w trybie nieszyfrowanym. Więcej informacji można znaleźć w sekcji pozycje agenta. Parametr 'Ports' może mieć jedną z następujących wartości: Pojedynczy port: 22 Zakres portów: 22-45 Lista: 22-45,55,60-70 Od Zabbix 7.0 wszystkie sprawdzenia usług są wykonywane asynchronicznie, z wyjątkiem sprawdzeń LDAP. Od Zabbix 7.0 sprawdzanie HTTP/HTTPS odbywa się za pomocą libcurl. Jeśli serwer/proxy Zabbix został skompilowany bez libcurl, sprawdzenia HTTP będą działać jak w poprzednich wersjach (tj. jako sprawdzenia TCP), ale sprawdzenia HTTPS nie będą działać.
Device uniqueness criteria	Kryteria unikalności mogą być następujące: **IP address** - nie przetwarzaj wielu urządzeń z pojedynczym adresem IP. Jeśli urządzenie o tym samym adresie IP już istnieje, zostanie uznane za już wykryte i nowy host nie zostanie dodany. **<discovery check>** - sprawdzenie Zabbix agent lub SNMP agent. Należy pamiętać, że kryteria unikalności używane podczas wykrywania nie są tym samym co identyfikacja host w systemie podczas wykonywania akcji. Kryteria unikalności podczas wykrywania określają, czy dwa lub więcej wykrytych urządzeń jest takich samych (lub różnych), natomiast w Zabbix kryterium identyfikacji host jest tylko adres IP (zobacz Host creation).
Host name	Ustaw techniczną nazwę host utworzonego hosta, używając: **DNS name** - nazwa DNS (domyślnie) **IP address** - adres IP **<discovery check>** - otrzymana wartość tekstowa sprawdzenia wykrywania (np. Zabbix agent, sprawdzenie SNMP agent) Zobacz też: Host naming.
Visible name	Ustaw widoczną nazwę host utworzonego hosta, używając: **Host name** - techniczna nazwa host (domyślnie) **DNS name** - nazwa DNS **IP address** - adres IP **<discovery check>** - otrzymana wartość tekstowa sprawdzenia wykrywania (np. Zabbix agent, sprawdzenie SNMP agent) Zobacz też: Host naming.
Enabled	Gdy pole wyboru jest zaznaczone, reguła jest aktywna i będzie wykonywana przez serwer Zabbix. Jeśli nie jest zaznaczone, reguła nie jest aktywna. Nie będzie wykonywana.

Przekroczenie limitu deskryptorów plików

W przypadku dużej liczby równoczesnych sprawdzeń możliwe jest wyczerpanie limitu deskryptorów plików dla discovery manager.

Liczba deskryptorów plików wymaganych do wykrywania odpowiada liczbie discovery workers * 1000. Domyślnie istnieje 5 discovery workers, natomiast miękki limit systemu wynosi około 1024.

Jeśli ten limit zostanie osiągnięty, Zabbix zmniejszy domyślną liczbę równoczesnych sprawdzeń na typ dla każdego worker i zapisze ostrzeżenie do pliku dziennika. Jeśli jednak użytkownik ustawił wyższą wartość dla *Maksymalna liczba równoczesnych sprawdzeń na typ* niż wartość obliczona przez Zabbix, Zabbix użyje wartości zdefiniowanej przez użytkownika dla jednego worker.

Przykład z życia

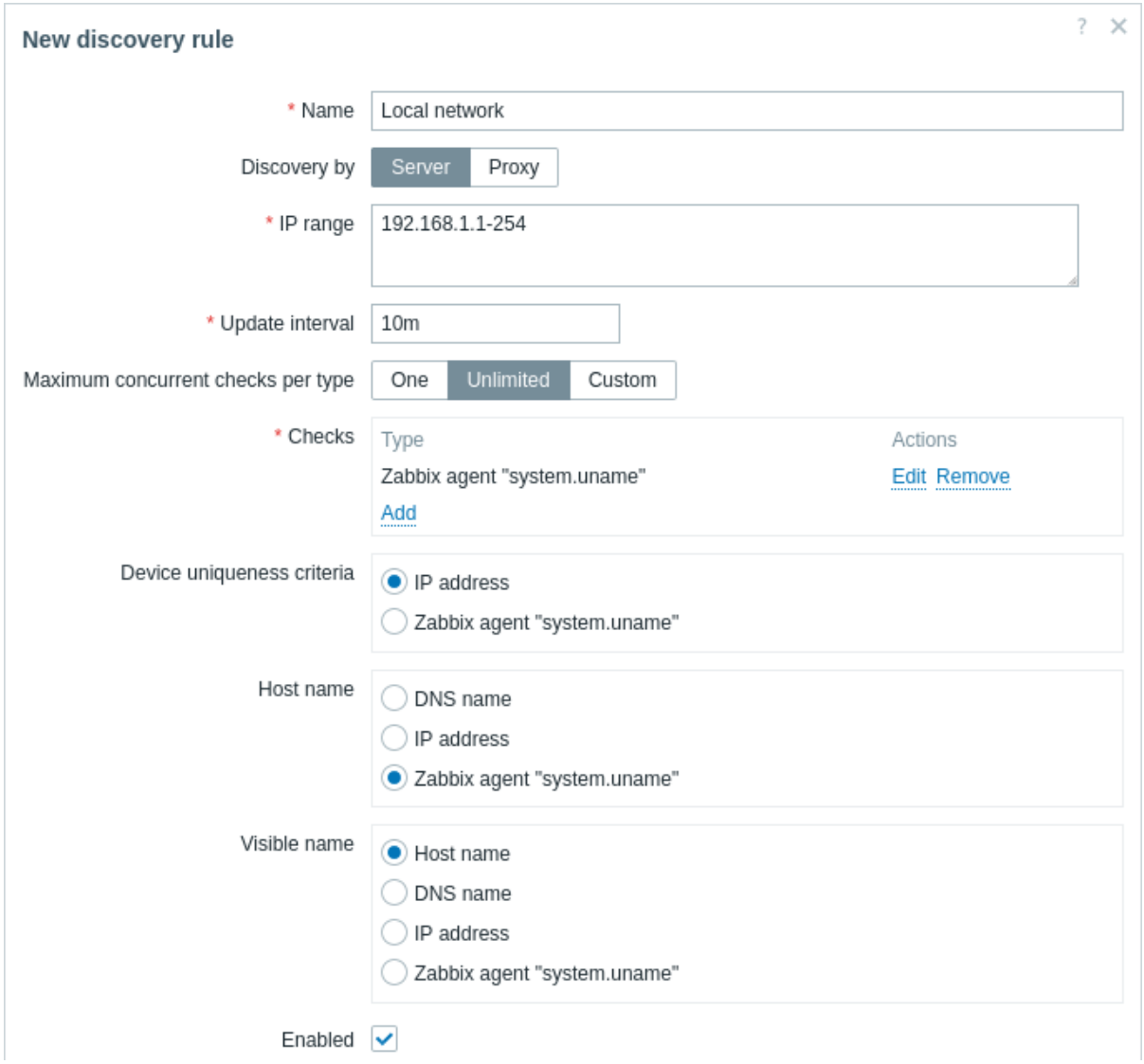
W tym przykładzie chcemy skonfigurować wykrywanie sieci dla sieci lokalnej o zakresie adresów IP 192.168.1.1-192.168.1.254.

W naszym scenariuszu chcemy:

- wykrywać te hosty, na których działa agent Zabbix
- uruchamiać wykrywanie co 10 minut
- dodać host do monitorowania, jeśli czas działania hosta wynosi więcej niż 1 godzinę
- usuwać hosty, jeśli czas niedostępności hosta przekracza 24 godziny
- dodawać hosty z systemem Linux do grupy ”Linux servers”
- dodawać hosty z systemem Windows do grupy ”Windows servers”
- używać szablonu *Linux* dla hostów z systemem Linux
- używać szablonu *Windows* dla hostów z systemem Windows

Krok 1

Definiowanie reguły wykrywania sieci dla naszego zakresu adresów IP.



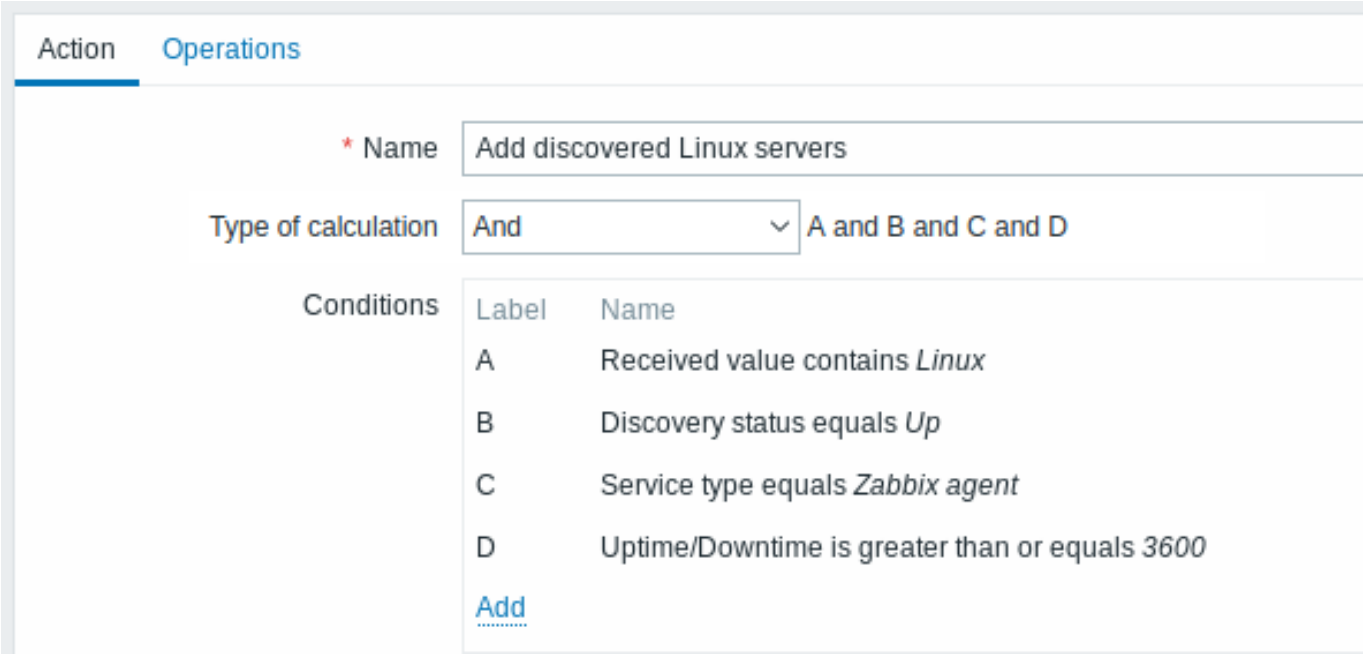
Zabbix spróbuje wykryć hosty w zakresie adresów IP 192.168.1.1-192.168.1.254, łącząc się z agentami Zabbix i pobierając wartość z klucza **system.uname**. Wartość otrzymana od agent może zostać użyta do nadania nazw hostom, a także do zastosowania różnych akcji dla różnych systemów operacyjnych. Na przykład powiązać serwery Windows z szablonem *Windows*, a serwery Linux z szablonem *Linux*.

Reguła będzie wykonywana co 10 minut.

Po dodaniu tej reguły Zabbix automatycznie rozpocznie wykrywanie i generowanie zdarzeń opartych na wykrywaniu do dalszego przetwarzania.

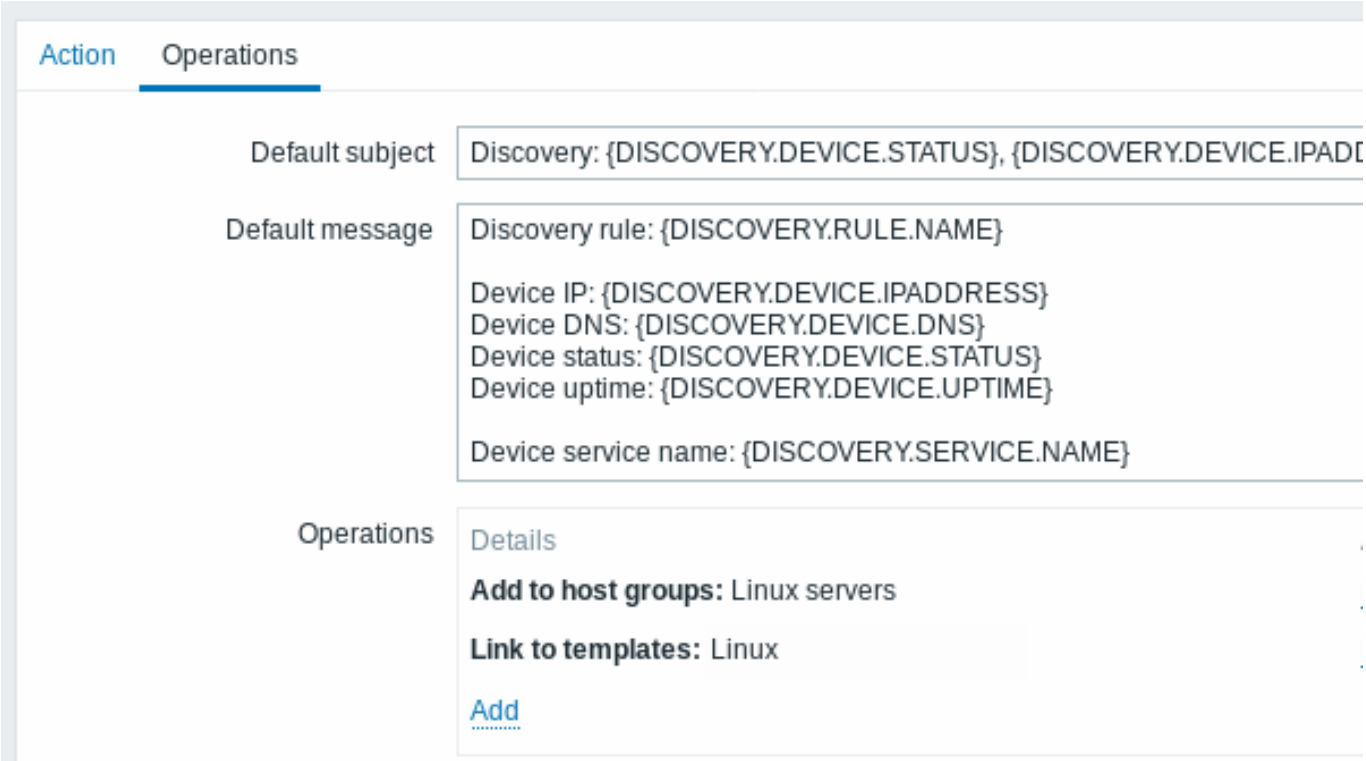
Krok 2

Definiowanie akcji wykrywania do dodawania wykrytych serwerów Linux do odpowiedniej grupy/szablonu.



Akcja zostanie uruchomiona, jeśli:

- usługa „Zabbix agent" ma stan „up"
- wartość system.uname (klucz Zabbix agent użyty przez nas w definicji reguły) zawiera „Linux"
- czas działania wynosi 1 godzinę (3600 sekund) lub więcej

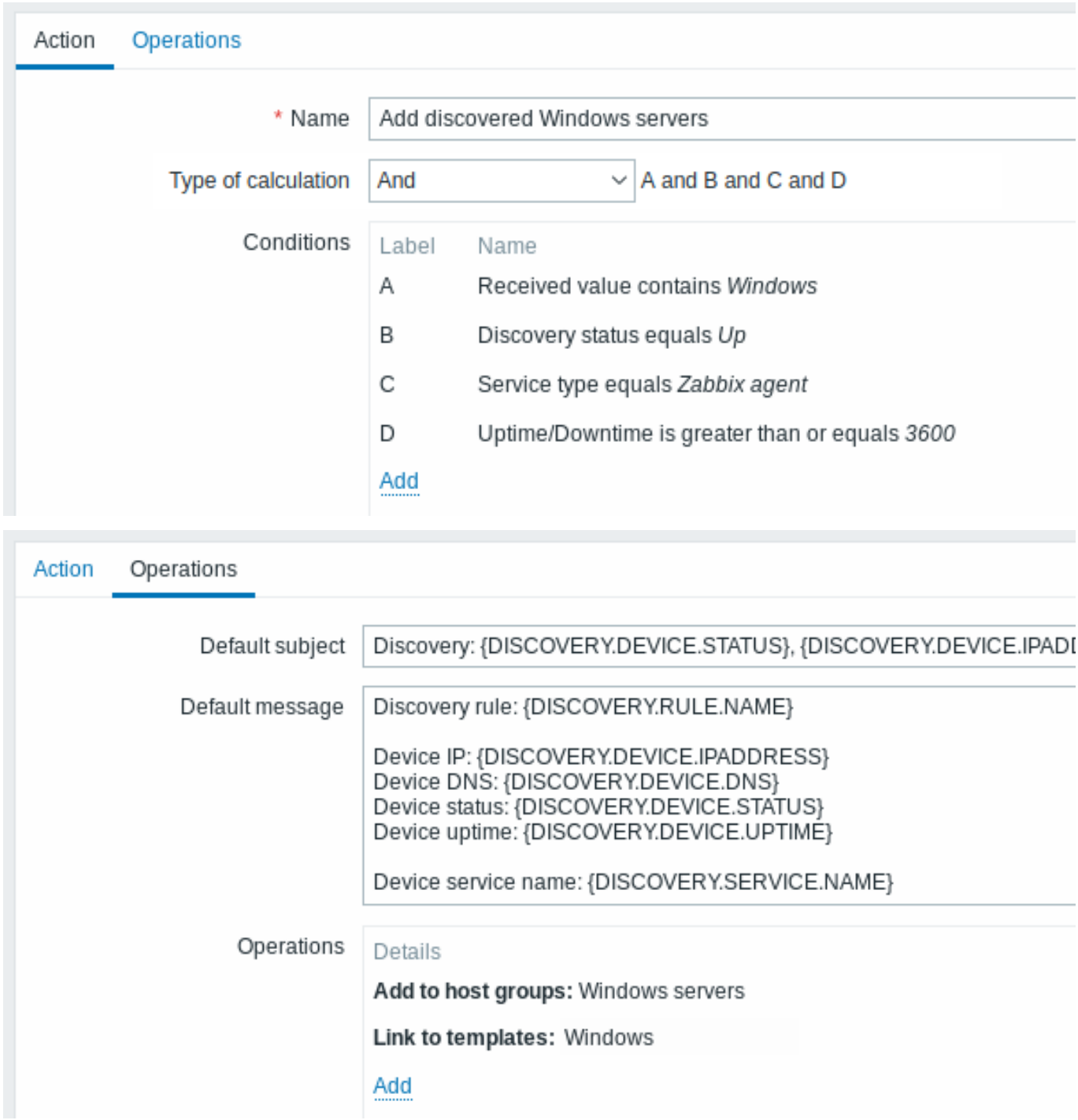


Akcja wykona następujące operacje:

- doda wykryty host do grupy „Linux servers" (a także doda host, jeśli nie został dodany wcześniej)
- podłączy host do szablonu *Linux*. Zabbix automatycznie rozpocznie monitorowanie hosta przy użyciu pozycji i wyzwalaczy z szablonu „Linux".

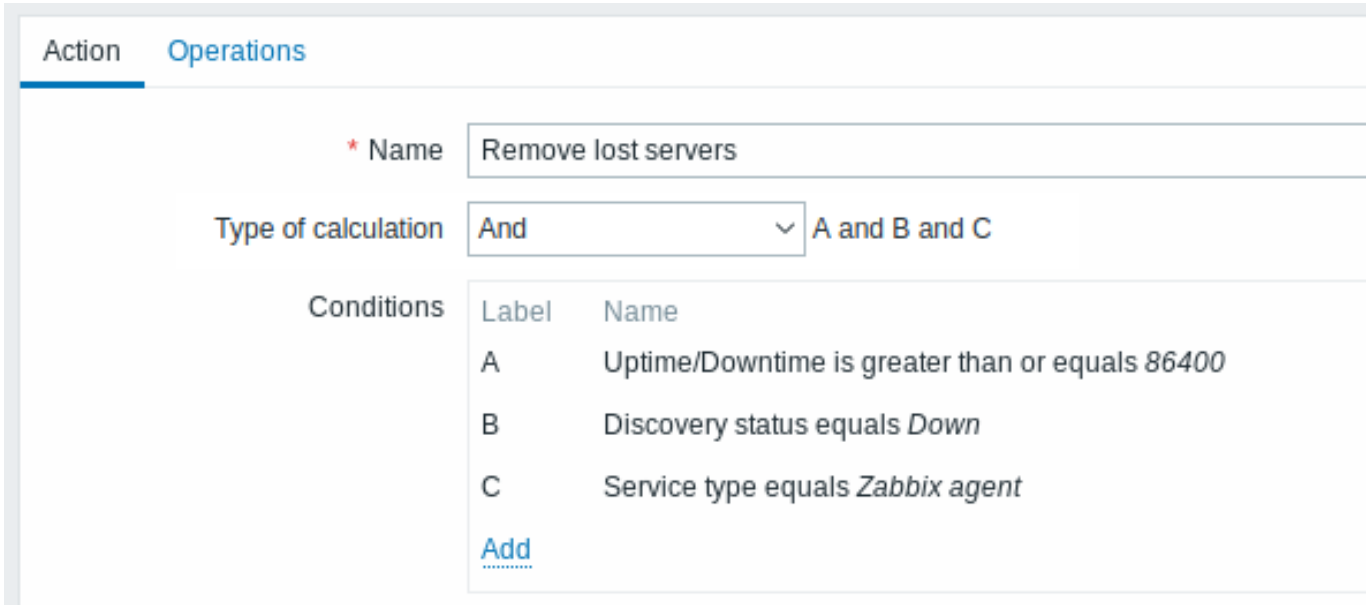
Krok 3

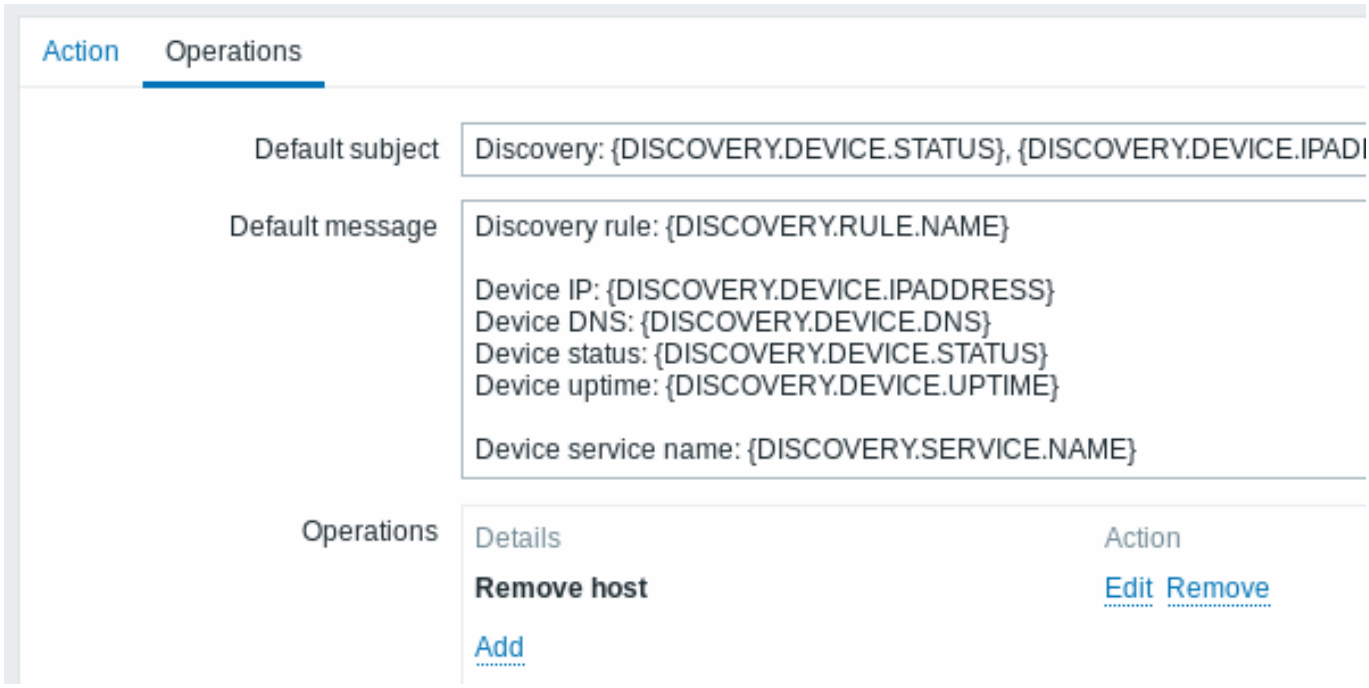
Definiowanie akcji wykrywania służącej do dodawania wykrytych serwerów Windows do odpowiedniej grupy/szablonu.



Krok 4

Definiowanie akcji wykrywania do usuwania utraconych serwerów.





Serwer zostanie usunięty, jeśli usługa „Zabbix agent” będzie w stanie „down” przez ponad 24 godziny (86400 sekund).

Automatyczna rejestracja aktywnego agenta

Omówienie

Możliwe jest zezwolenie na automatyczną rejestrację aktywnego agenta Zabbix, po czym serwer może rozpocząć jego monitorowanie. W ten sposób nowe hosty można dodawać do monitorowania bez ręcznej konfiguracji na serwerze.

Automatyczna rejestracja może nastąpić, gdy wcześniej nieznany aktywny agent zażąda sprawdzeń.

Ta funkcja może być bardzo przydatna do automatycznego monitorowania nowych węzłów chmurowych. Gdy tylko pojawi się nowy węzeł w Zabbix Cloud, automatycznie rozpocznie się zbieranie danych o wydajności i dostępności hosta.

Automatyczna rejestracja aktywnego agenta obsługuje również monitorowanie dodanych hostów za pomocą pasywnych sprawdzeń. Gdy aktywny agent zażąda sprawdzeń, a w pliku konfiguracyjnym zdefiniowano parametry konfiguracyjne `ListenIP` lub `ListenPort`, są one przekazywane do serwera. Jeśli określono wiele adresów IP, do serwera wysyłany jest pierwszy z nich.

Serwer, podczas dodawania nowego hosta zarejestrowanego automatycznie, używa otrzymanego adresu IP i portu do skonfigurowania agenta. Jeśli nie zostanie odebrana wartość adresu IP, używany jest adres zastosowany dla połączenia przychodzącego. Jeśli nie zostanie odebrana wartość portu, używany jest 10050.

Możliwe jest określenie, że host powinien zostać zarejestrowany automatycznie z nazwą DNS jako domyślnym interfejsem agenta.

Automatyczna rejestracja jest ponawiana:

- jeśli zmienią się informacje o metadanych hosta:
  - z powodu zmiany HostMetadata i ponownego uruchomienia agenta
  - z powodu zmiany wartości zwracanej przez HostMetadataItem
- dla hostów utworzonych ręcznie, gdy brakuje metadanych
- jeśli host zostanie ręcznie zmieniony tak, aby był monitorowany przez inny proxy Zabbix
- jeśli automatyczna rejestracja tego samego hosta pochodzi z nowego proxy Zabbix

Interwał heartbeat automatycznej rejestracji aktywnego agenta dla serwera Zabbix i proxy Zabbix wynosi 120 sekund. W związku z tym, jeśli wykryty host zostanie usunięty, automatyczna rejestracja zostanie ponowiona po 120 sekundach.

Konfiguracja

Określ serwer

Upewnij się, że serwer Zabbix został określony w pliku konfiguracyjnym agenta (`zabbix_agentd.conf`):

```
ServerActive=10.0.0.1
```

Jeśli w pliku `zabbix_agentd.conf` nie określisz jawnie wartości *Hostname*, serwer użyje nazwy systemowej hosta, na którym działa agent, do nazwania hosta. Nazwę systemową hosta w systemie Linux można uzyskać, uruchamiając polecenie `hostname`.

Jeśli wartość *Hostname* zostanie określona w konfiguracji agenta Zabbix jako rozdzielana przecinkami lista hostów, hosty zostaną utworzone dla wszystkich wymienionych nazw hostów.

Po wprowadzeniu zmian w pliku konfiguracyjnym uruchom ponownie agenta.

Działanie na potrzeby automatycznej rejestracji aktywnego agenta

Gdy serwer otrzyma od agenta żądanie automatycznej rejestracji, wywołuje działanie. Na potrzeby automatycznej rejestracji agenta należy skonfigurować działanie ze źródłem zdarzeń „Autorejestracja".

Note:
Skonfigurowanie wykrywania sieci nie jest wymagane, aby aktywne agenty mogły się automatycznie rejestrować.

W frontendzie Zabbix:

1. Przejdź do *Alerty* > *Działania*.
2. Wybierz *Działania automatycznej rejestracji*.
3. Kliknij *Utwórz działanie*.
4. Na karcie *Działanie* nadaj działaniu nazwę.
5. Opcjonalnie określ warunki. W warunkach dotyczących nazwy hosta/metadanych hosta można użyć dopasowania podciągu lub wyrażenia regularnego. Jeśli zamierzasz użyć typu warunku *Metadane hosta*, zapoznaj się z sekcją Używanie metadanych hosta.
6. Na karcie *Operacje* dodaj odpowiednie operacje, takie jak *Dodaj host*, *Dodaj do grupy hostów* (na przykład *Wykryte hosty*), *Połącz szablony* itd.

Note:
Jeśli hosty, które będą automatycznie rejestrowane, prawdopodobnie będą obsługiwane wyłącznie w trybie aktywnego monitorowania (na przykład hosty odseparowane zaporą sieciową od serwera Zabbix), możesz utworzyć specjalny szablon, taki jak *Template_Linux-active*, i połączyć go z tymi hostami.

Utworzone hosty są dodawane do grupy *Wykryte hosty* (domyślnie; można to skonfigurować w *Administracja* > *Ogólne* > *Inne*). Jeśli chcesz, aby hosty były dodawane do innej grupy, dodaj operację *Usuń z grupy hostów* (określając „Wykryte hosty"), a także operację *Dodaj do grupy hostów* (określając inną grupę hostów), ponieważ host musi należeć do grupy hostów.

Bezpieczna automatyczna rejestracja

Bezpieczna automatyczna rejestracja jest możliwa po skonfigurowaniu uwierzytelniania opartego na PSK z szyfrowanymi połączeniami.

Poziom szyfrowania jest konfigurowany globalnie w sekcji *Administracja* > *Ogólne* > *Automatyczna rejestracja*. Można wybrać brak szyfrowania, szyfrowanie TLS z uwierzytelnianiem PSK albo obie opcje (dzięki czemu niektóre hosty mogą rejestrować się bez szyfrowania, a inne za pośrednictwem szyfrowanego połączenia).

Uwierzytelnianie za pomocą PSK jest weryfikowane przez serwer Zabbix przed dodaniem hosta. Jeśli weryfikacja zakończy się pomyślnie, host zostaje dodany, a ustawienie *Połączenia z/do hosta* zostaje ustawione wyłącznie na „PSK", z identyfikatorem i kluczem wstępnie współdzielonymi takimi samymi jak w globalnym ustawieniu automatycznej rejestracji.

Attention:
Aby zapewnić bezpieczeństwo automatycznej rejestracji w instalacjach korzystających z proxy, należy włączyć szyfrowanie między serwerem Zabbix a proxy.

Używanie DNS jako domyślnego interfejsu

Parametry konfiguracji HostInterface i HostInterfaceItem configuration parameters umożliwiają określenie niestandardowej wartości interfejsu hosta podczas automatycznej rejestracji.

Są one szczególnie przydatne, jeśli host powinien zostać automatycznie zarejestrowany z nazwą DNS jako domyślnym interfejsem agenta zamiast jego adresem IP. W takim przypadku nazwa DNS powinna zostać określona lub zwrócona jako wartość jednego z parametrów HostInterface lub HostInterfaceItem. Jeśli wartość jednego z tych parametrów ulegnie zmianie — na przykład z adresu IP na nazwę DNS lub odwrotnie — domyślny interfejs automatycznie zarejestrowanego hosta zostanie odpowiednio zaktualizowany. Aktualizacja ta jest stosowana do istniejącego hosta, a nie przez utworzenie nowego. Aby wysłać nową wartość, należy ponownie uruchomić agenta, aby ponownie zainicjował proces automatycznej rejestracji.

Note:
Jeśli parametry HostInterface lub HostInterfaceItem nie są skonfigurowane, używany jest parametr listen_dns. Wartość ta jest określana przez wykonanie odwrotnego wyszukiwania DNS adresu IP agenta. Jeśli odwrotne wyszukiwanie DNS nie jest prawidłowo skonfigurowane lub zwraca nieprawidłową nazwę, może to spowodować nieprawidłową lub nieudaną automatyczną rejestrację z powodu nieprawidłowej wartości interfejsu.

Korzystanie z metadanych hosta

Gdy agent wysyła do serwera żądanie automatycznej rejestracji, przesyła swoją nazwę hosta. W niektórych przypadkach (na przykład w przypadku węzłów chmury Amazon) nazwa hosta nie wystarcza serwerowi Zabbix do rozróżnienia wykrytych hostów. Metadane hosta mogą być opcjonalnie używane do przesyłania dodatkowych informacji z agenta do serwera.

Metadane hosta konfiguruje się w pliku konfiguracyjnym agenta (`zabbix_agentd.conf`). Metadane hosta można określić w pliku konfiguracyjnym na 2 sposoby:

```
HostMetadata
HostMetadataItem
```

Opis opcji znajduje się w powyższym odnośniku.

Parametr HostMetadataItem może zwrócić maksymalnie 65535 punktów kodowych UTF-8. Dłuższa wartość zostanie obcięta.

Należy pamiętać, że w systemie MySQL efektywna maksymalna długość w znakach będzie mniejsza, jeśli zwrócona wartość zawiera znaki wielobajtowe. Na przykład wartość zawierająca wyłącznie znaki 3-bajtowe będzie ograniczona do 21844 znaków łącznie, natomiast wartość zawierająca wyłącznie znaki 4-bajtowe będzie ograniczona do 16383 symboli.

Attention:
Próba automatycznej rejestracji ma miejsce za każdym razem, gdy aktywny agent wysyła do serwera żądanie odświeżenia aktywnych kontroli. Opóźnienie między żądaniami określa parametr RefreshActiveChecks agenta. Pierwsze żądanie jest wysyłane natychmiast po ponownym uruchomieniu agenta.

Przykłady

Autorejestracja według systemu operacyjnego przy użyciu HostMetadata

Załóżmy, że chcesz, aby hosty były automatycznie rejestrowane przez serwer Zabbix. W swojej sieci masz aktywne agenty Zabbix (zobacz sekcję „Configuration” powyżej). W sieci znajdują się hosty z systemem Windows i Linux, a w frontendzie Zabbix masz dostępne szablony „Linux by Zabbix agent” i „Windows by Zabbix agent”. Dlatego podczas rejestracji hosta chcesz, aby do rejestrowanego hosta został zastosowany odpowiedni szablon Linux/Windows. Domyślnie podczas autorejestracji do serwera wysyłana jest tylko nazwa hosta, co może nie być wystarczające. Aby mieć pewność, że do hosta zostanie zastosowany właściwy szablon, należy użyć metadanych hosta.

Konfiguracja frontendu

Pierwszą rzeczą, którą należy zrobić, jest skonfigurowanie frontendu. Utwórz 2 akcje. Pierwsza akcja:

- Nazwa: Automatyczna rejestracja hosta Linux
- Warunki: Metadane hosta zawierają *Linux*
- Operacje: Podłącz szablony: Linux by Zabbix agent

Note:
W tym przypadku można pominąć operację „Dodaj host”. Podłączenie szablonu do hosta wymaga najpierw dodania hosta, więc serwer zrobi to automatycznie.

Druga akcja:

- Nazwa: Automatyczna rejestracja hosta Windows
- Warunki: Metadane hosta zawierają *Windows*
- Operacje: Podłącz szablony: Windows by Zabbix agent

Konfiguracja agenta

Teraz należy skonfigurować agenty. Dodaj następujący wiersz do plików konfiguracyjnych agenta:

```
HostMetadataItem=system.uname
```

W ten sposób masz pewność, że metadane hosta będą zawierać „Linux” lub „Windows” w zależności od hosta, na którym uruchomiony jest agent. Przykład metadanych hosta w tym przypadku:

```
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
```

Nie zapomnij zrestartować agenta po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym.

Używanie HostMetadata do kontrolowania autorejestracji i szablonów

Krok 1 — Chroń rejestrację za pomocą HostMetadata

Użycie metadanych hosta pozwala zapewnić podstawową ochronę przed rejestracją niepożądanych hostów.

Konfiguracja frontendu

Utwórz akcję we frontendzie, używając trudnego do odgadnięcia tajnego kodu, aby nie dopuścić niepożądanych hostów:

- Nazwa: Akcja autorejestracji Linux
- Warunki:
  - Typ obliczeń: AND
  - Warunek (A): Metadane hosta zawierają //Linux//
  - Warunek (B): Metadane hosta zawierają //21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae//
- Operacje:
  - Wyślij wiadomość do użytkowników: Admin przez wszystkie media
  - Dodaj do grup hostów: Serwery Linux
  - Podłącz szablony: Linux by Zabbix agent

Należy pamiętać, że sama ta metoda nie zapewnia silnej ochrony, ponieważ dane są przesyłane otwartym tekstem. Aby zmiany zaczęły obowiązywać natychmiast, wymagane jest przeładowanie pamięci podręcznej konfiguracji.

Konfiguracja agenta

Dodaj następujący wiersz do pliku konfiguracyjnego agenta:

```
HostMetadata=Linux    21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

gdzie „Linux” to platforma, a pozostała część ciągu to trudny do odgadnięcia tajny tekst.

Nie zapomnij zrestartować agenta po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym.

Krok 2 — Dodaj szablon do zarejestrowanego hosta

Możliwe jest dodanie dodatkowych szablonów do już zarejestrowanego hosta. W takim przypadku szablon MySQL by Zabbix agent zostanie podłączony tylko do hostów, których HostMetadata zawiera token `MySQL`.

Konfiguracja frontend

Zaktualizuj akcję we frontend:

- Nazwa: Akcja autorejestracji Linux
- Warunki:
  - Typ obliczania: AND
  - Warunek (A): Metadane host zawierają Linux
  - Warunek (B): Metadane host zawierają 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
  - Warunek (C): Metadane host zawierają MySQL
- Operacje:
  - Wyślij wiadomość do użytkowników: Admin przez wszystkie media

- Dodaj do grup hostów: Serwery Linux
- Podłącz szablony: Linux by Zabbix agent
- Podłącz szablony: MySQL by Zabbix Agent

Konfiguracja agenta

Zaktualizuj następną linię w pliku konfiguracyjnym agenta:

```
HostMetadata=MySQL on Linux 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

Nie zapomnij zrestartować agenta po wprowadzeniu jakichkolwiek zmian w pliku konfiguracyjnym.

Odkrywanie niskiego poziomu

Omówienie Low-level discovery (LLD) automatycznie wykrywa encje (np. systemy plików, interfejsy sieciowe) na host i tworzy odpowiadające im pozycje, wyzwalacze i wykresy bez ręcznej konfiguracji dla każdej z nich.

Aby używać LLD, tworzysz **regułę wykrywania**, która zbiera dane JSON opisujące encje, oraz **prototypy** (pozycja, wyzwalacz i wykres) dla każdej encji.

JSON zwracany przez regułę musi być tablicą obiektów, gdzie każdy obiekt reprezentuje jedną wykrytą encję za pomocą par klucz-wartość. Na przykład reguła wykrywania z `net.if.discovery` może zwrócić:

```
[
  {"{#IFNAME}": "lo"},
  {"{#IFNAME}": "eth0"}
]
```

Odpowiedni zestaw pozycji, wyzwalaczy i wykresów jest następnie tworzony na podstawie prototypów — po jednym dla każdego interfejsu (`lo` i `eth0`).

Jeśli obiekty zawierają klucze w formacie `{#MACRO}`, te makra są używane bezpośrednio w prototypach. Jeśli nie, lub jeśli wymagane są dodatkowe/własne makra, można je zdefiniować ręcznie i mapować do pól JSON za pomocą JSONPath.

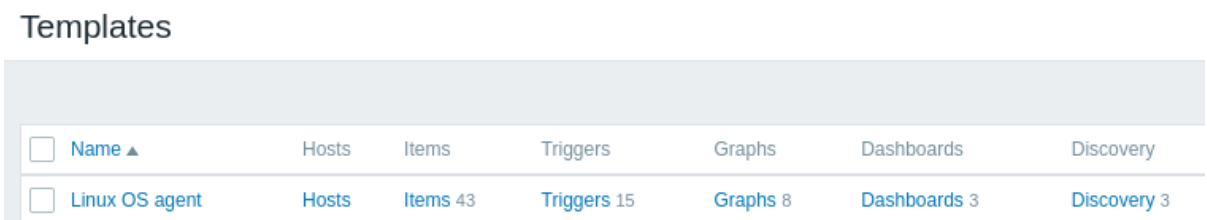
LLD może również tworzyć hosty (np. dla maszyn wirtualnych wykrytych na hypervisorze) oraz obsługuje zagnieżdżone reguły wykrywania dla wielopoziomowego wykrywania.

Encje, które nie są już wykrywane, mogą być automatycznie wyłączane lub usuwane.

Konfigurowanie wykrywania niskiego poziomu Przedstawimy wykrywanie niskiego poziomu na przykładzie wykrywania systemu plików.

Aby skonfigurować wykrywanie, wykonaj następujące czynności:

1. Przejdź do *Data collection* > *Templates* lub *Hosts*.
2. Kliknij *Discovery* w wierszu odpowiedniego szablonu/hosta.



3. Kliknij *Create discovery rule* w prawym górnym rogu ekranu.
4. Wypełnij formularz reguły wykrywania, podając wymagane szczegóły.

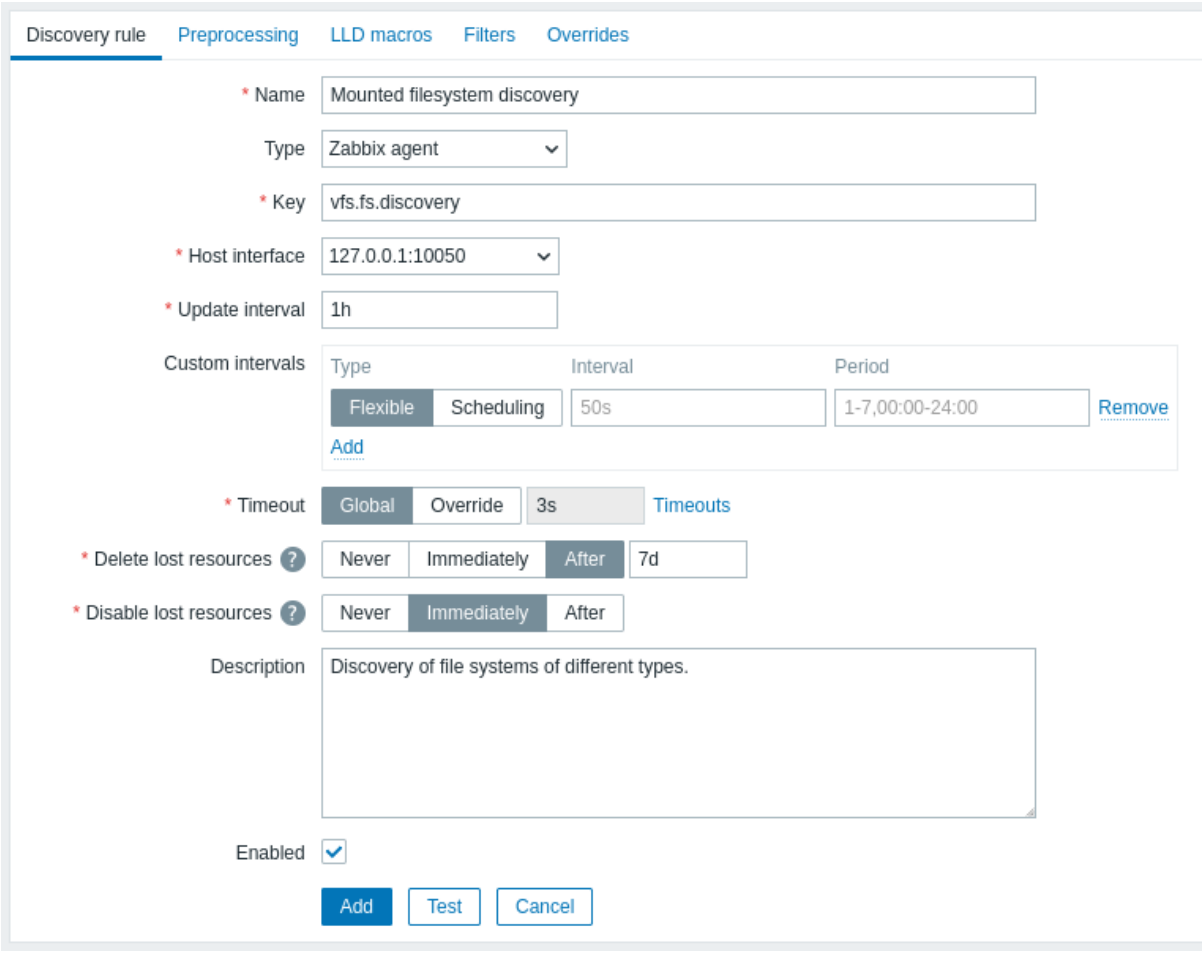
Reguła wykrywania

Formularz reguły wykrywania zawiera pięć kart, przedstawiających od lewej do prawej przepływ danych podczas wykrywania:

- *Reguła wykrywania* - określ przede wszystkim wbudowaną pozycję lub niestandardowy skrypt używany do pobierania danych wykrywania.
- *Przetwarzanie wstępne* - zastosuj przetwarzanie wstępne do wykrytych danych.
- *Makra LLD* - umożliwiają wyodrębnianie wartości makr do użycia w wykrytych pozycjach, wyzwalaczach itd.
- *Filtry* - umożliwiają filtrowanie wykrytych wartości.

- *Zastąpienia* - umożliwiają modyfikowanie prototypów pozycji, wyzwalaczy, wykresów lub hostów podczas stosowania ich do określonych wykrytych obiektów.

Karta **Reguła wykrywania** zawiera klucz pozycji używany do wykrywania (a także niektóre ogólne atrybuty reguły wykrywania):



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametr	Opis
Nazwa	Nazwa reguły wykrywania.
Typ	Typ sprawdzenia używanego do wykrywania. W tym przykładzie używamy typu pozycji *Zabbix agent*. Reguła wykrywania może również być pozycją zależną, zależną od zwykłej pozycji. Nie może zależeć od innej reguły wykrywania. W przypadku pozycji zależnej wybierz odpowiedni typ (*Pozycja zależna*) i określ pozycję nadrzędną w polu „Pozycja nadrzędna”. Pozycja nadrzędna musi istnieć.
Klucz	Wprowadź klucz pozycji wykrywania (do 2048 znaków). Możesz na przykład użyć wbudowanego klucza pozycji „vfs.fs.discovery”, aby zwrócić ciąg JSON zawierający listę systemów plików obecnych na komputerze, ich typy i opcje montowania. Pamiętaj, że inną opcją wykrywania systemów plików jest użycie wyników wykrywania za pomocą klucza agenta „vfs.fs.get” (zobacz przykład).

Parametr	Opis
Interwał aktualizacji	To pole określa, jak często serwer Zabbix wykonuje wykrywanie. Na początku, podczas konfigurowania wykrywania systemów plików, możesz ustawić krótki interwał, ale gdy upewnisz się, że wszystko działa, możesz ustawić 30 minut lub więcej, ponieważ systemy plików zwykle nie zmieniają się często. Obsługiwane są przyrostki czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: Interwał aktualizacji można ustawić na „0” tylko wtedy, gdy istnieją interwały niestandardowe z wartością różną od zera. Jeśli ustawiono „0” i istnieje interwał niestandardowy (elastyczny lub zaplanowany) z wartością różną od zera, pozycja będzie odpytywana w czasie trwania interwału niestandardowego. Nowe reguły wykrywania zostaną sprawdzone w ciągu 60 sekund od ich utworzenia, chyba że mają ustawiony interwał aktualizacji typu Harmonogram lub Elastyczny, a *Interwał aktualizacji* ma wartość 0. *Uwaga*: W przypadku istniejącej reguły wykrywania wykrywanie można wykonać natychmiast, klikając przycisk *Wykonaj teraz*.
Interwały niestandardowe	Możesz utworzyć niestandardowe reguły sprawdzania pozycji: **Elastyczny** - utwórz wyjątek od *Interwału aktualizacji* (interwał o innej częstotliwości) **Harmonogram** - utwórz niestandardowy harmonogram odpytywania. Szczegółowe informacje znajdziesz w sekcji Interwały niestandardowe.
Limit czasu	Ustaw limit czasu sprawdzania wykrywania. Wybierz opcję limitu czasu: **Globalny** - używany jest limit czasu proxy/globalny (wyświetlany w wyszarzonym polu *Limit czasu*); **Zastąpienie** - używany jest niestandardowy limit czasu (ustawiony w polu *Limit czasu*; dozwolony zakres: 1 - 600 s). Obsługiwane są przyrostki czasu, np. 30s, 1m, oraz makra użytkownika. Kliknięcie łącza *Limity czasu* umożliwia skonfigurowanie limitów czasu proxy lub globalnych (jeśli proxy nie jest używany). Łącze *Limity czasu* jest widoczne tylko dla użytkowników typu *Super admin* z uprawnieniami do sekcji frontend *Administracja* > *Ogólne* lub *Administracja* > *Proxy*.
Usuń utracone zasoby	Określ, jak szybko wykryta jednostka zostanie usunięta, gdy jej status wykrywania zmieni się na „Już niewykrywana”: *Nigdy* - nie zostanie usunięta; *Natychmiast* - zostanie usunięta natychmiast; *Po* - zostanie usunięta po określonym czasie. Wartość musi być większa niż wartość *Wyłącz utracone zasoby*. Obsługiwane są przyrostki czasu, np. 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: Nie zaleca się używania opcji „Natychmiast”, ponieważ zwykła nieprawidłowa edycja filtra może spowodować usunięcie jednostki wraz ze wszystkimi danymi historycznymi. Pamiętaj, że zasoby wyłączone ręcznie nie zostaną usunięte przez wykrywanie niskopoziomowe.
Wyłącz utracone zasoby	Określ, jak szybko wykryta jednostka zostanie wyłączona, gdy jej status wykrywania zmieni się na „Już niewykrywana”: *Nigdy* - nie zostanie wyłączona; *Natychmiast* - zostanie wyłączona natychmiast; *Po* - zostanie wyłączona po określonym czasie. Wartość powinna być większa niż interwał aktualizacji reguły wykrywania. Pamiętaj, że zasoby wyłączone automatycznie zostaną ponownie włączone, jeśli zostaną ponownie wykryte przez wykrywanie niskopoziomowe. Zasoby wyłączone ręcznie nie zostaną ponownie włączone po ponownym wykryciu. To pole nie jest wyświetlane, jeśli opcja *Usuń utracone zasoby* jest ustawiona na „Natychmiast”. Obsługiwane są przyrostki czasu, np. 2h, 1d. Obsługiwane są makra użytkownika.
Opis	Wprowadź opis.
Włączone	Jeśli opcja jest zaznaczona, reguła będzie przetwarzana.

Note:
Historia reguły wykrywania nie jest zachowywana.

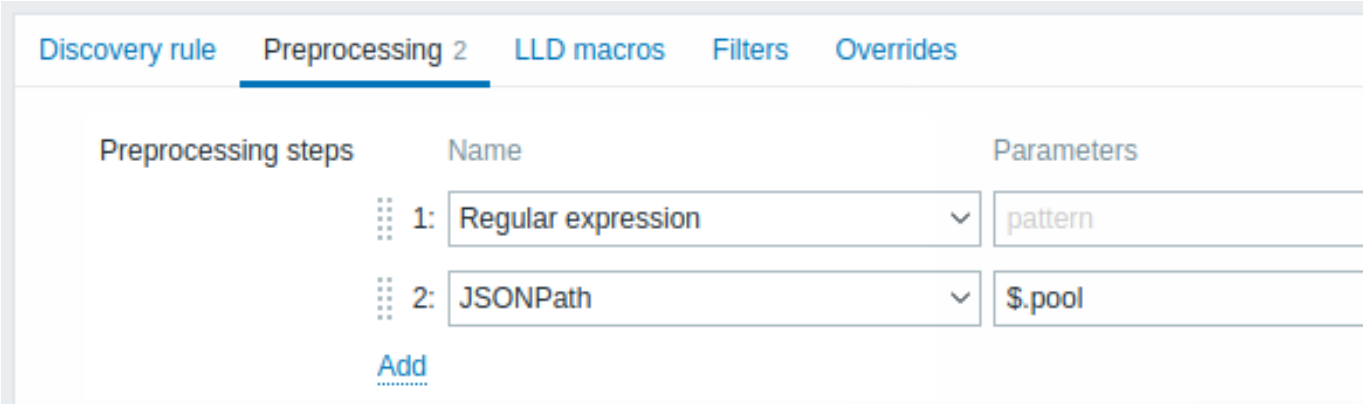
Wstępne przetwarzanie

Karta **Wstępne przetwarzanie** umożliwia zdefiniowanie reguł transformacji stosowanych do wyniku wykrywania. W tym kroku można zastosować jedną lub kilka transformacji. Transformacje są wykonywane w kolejności, w której zostały zdefiniowane. Całe

wstępne przetwarzanie jest wykonywane przez serwer Zabbix.

Zobacz także:

- Szczegóły wstępnego przetwarzania
- Testowanie wstępnego przetwarzania



Typ		
	Transformacja	Opis
Tekst		
	Wyrażenie regularne	Dopasuj otrzymaną wartość do wyrażenia regularnego <pattern> i zastąp wartość wyodrębnionym elementem <output>. Wyrażenie regularne obsługuje wyodrębnianie maksymalnie 10 przechwyconych grup za pomocą sekwencji \N. Parametry: **pattern** - wyrażenie regularne **output** - szablon formatowania wyniku. Sekwencja ucieczki \N (gdzie N=1...9) jest zastępowana przez N-tą dopasowaną grupę. Sekwencja ucieczki \0 jest zastępowana przez dopasowany tekst. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
	Zastąp	Znajdź ciąg wyszukiwania i zastąp go innym ciągiem (lub niczym). Wszystkie wystąpienia ciągu wyszukiwania zostaną zastąpione. Parametry: **search string** - ciąg do znalezienia i zastąpienia, z uwzględnieniem wielkości liter (wymagany) **replacement** - ciąg, którym należy zastąpić ciąg wyszukiwania. Ciąg zastępujący może być również pusty, co umożliwia usunięcie znalezionego ciągu wyszukiwania. Można używać sekwencji ucieczki do wyszukiwania lub zastępowania znaków nowego wiersza, powrotu karetki, tabulatorów i spacji ”\n \r \t \s”; ukośnik odwrotny można zapisać jako ”\\” a sekwencje ucieczki można zapisać jako ”\\n”. Uciekanie znaków nowego wiersza, powrotu karetki i tabulatorów jest wykonywane automatycznie podczas wykrywania niskopoziomowego.
Dane struk-tu-ralne		
	JSONPath	Wyodrębnij wartość lub fragment z danych JSON za pomocą funkcji JSONPath. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.

Typ		

	XML XPath	Wyodrębnij wartość lub fragment z danych XML za pomocą funkcji XPath.
		Aby ta opcja działała, serwer Zabbix musi zostać skompilowany z obsługą libxml.
		Przykłady:
		`number(/document/item/value)` wyodrębni 10 z `<document><item><value>10</value></item></document>`
		`number(/document/item/@attribute)` wyodrębni 10 z `<document><item attribute="10"></item></document>`
		`/document/item` wyodrębni `<item><value>10</value></item>` z `<document><item><value>10</value></item></document>`
		Należy pamiętać, że przestrzenie nazw nie są obsługiwane.
		Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.

	CSV do JSON	Konwertuj dane pliku CSV do formatu JSON. Więcej informacji można znaleźć tutaj: Wstępne przetwarzanie CSV do JSON.
	XML do JSON	Konwertuj dane w formacie XML do formatu JSON. Więcej informacji można znaleźć tutaj: Reguły serializacji. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
SNMP		
	Wartość SNMP walk	Wyodrębnij wartość według określonej nazwy OID/MIB i zastosuj opcje formatowania: **Bez zmian** - zwróć Hex-STRING jako nieescapowany ciąg szesnastkowy (*uwaga*: wskazówki wyświetlania są nadal stosowane); **UTF-8 z Hex-STRING** - konwertuj Hex-STRING do ciągu UTF-8; **MAC z Hex-STRING** - konwertuj Hex-STRING do ciągu adresu MAC (w którym `' '` zostanie zastąpione przez `':'`); **Integer z BITS** - konwertuj pierwsze 8 bajtów ciągu bitów wyrażonego jako sekwencja znaków szesnastkowych (np. ”1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W ciągach bitów dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
	SNMP walk do JSON	Konwertuj wartości SNMP do formatu JSON. Określ nazwę pola w JSON oraz odpowiadającą jej ścieżkę OID SNMP. Wartości pól zostaną wypełnione wartościami z określonej ścieżki OID SNMP. Ten krok wstępnego przetwarzania można wykorzystać do wykrywania OID SNMP. Dostępne są podobne opcje formatowania wartości jak w kroku *Wartość SNMP walk*. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
	Wartość SNMP get	Zastosuj opcje formatowania do wartości SNMP get: **UTF-8 z Hex-STRING** - konwertuj Hex-STRING do ciągu UTF-8; **MAC z Hex-STRING** - konwertuj Hex-STRING do ciągu adresu MAC (w którym `' '` zostanie zastąpione przez `':'`); **Integer z BITS** - konwertuj pierwsze 8 bajtów ciągu bitów wyrażonego jako sekwencja znaków szesnastkowych (np. ”1A 2B 3C 4D”) na 64-bitową liczbę całkowitą bez znaku. W ciągach bitów dłuższych niż 8 bajtów kolejne bajty zostaną zignorowane. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
Skrypty niestandardowe		
	JavaScript	Wprowadź kod JavaScript w edytorze modalnym, który otwiera się po kliknięciu pola parametru lub ikony ołówka obok niego. Należy pamiętać, że dostępna długość kodu JavaScript zależy od używanej bazy danych. Więcej informacji można znaleźć tutaj: Wstępne przetwarzanie JavaScript

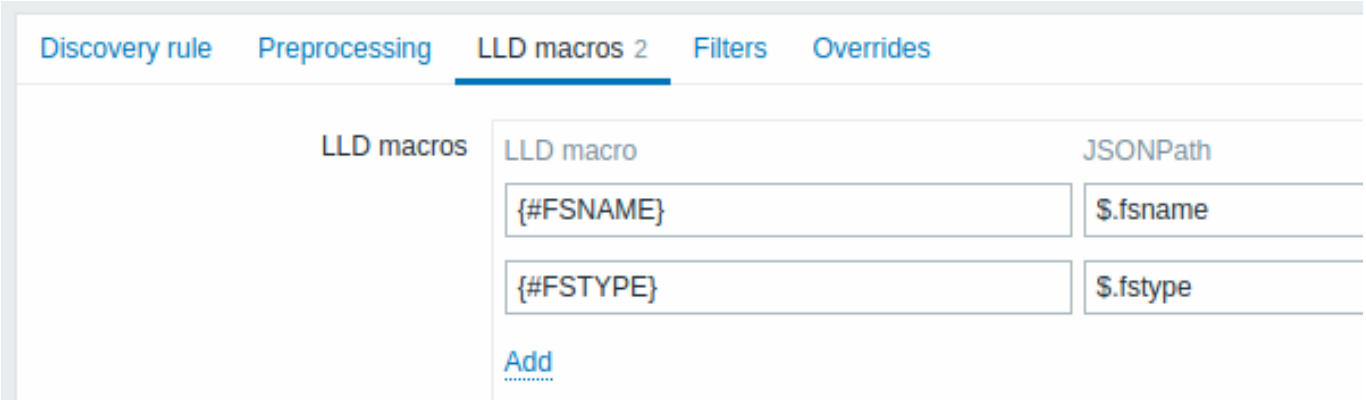
Typ	
Walidacja	
Nie pasuje do wyrażenia regularnego	Określ wyrażenie regularne, do którego wartość nie może pasować. Na przykład `Error:(.*?)\.` Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
Sprawdź błąd w JSON	Sprawdź komunikat o błędzie na poziomie aplikacji znajdujący się pod adresem JSONPath. Zatrzymaj przetwarzanie, jeśli sprawdzenie zakończy się powodzeniem, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie wartości, która znajdowała się przed tym krokiem wstępnego przetwarzania. Należy pamiętać, że te błędy usług zewnętrznych są zgłaszane użytkownikowi bez zmian, bez dodawania informacji o kroku wstępnego przetwarzania. Na przykład `$.errors`. Jeśli zostanie odebrany JSON taki jak `{"errors":"e1"}`, następny krok wstępnego przetwarzania nie zostanie wykonany. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
Sprawdź błąd w XML	Sprawdź komunikat o błędzie na poziomie aplikacji znajdujący się pod adresem XPath. Zatrzymaj przetwarzanie, jeśli sprawdzenie zakończy się powodzeniem, a komunikat nie jest pusty; w przeciwnym razie kontynuuj przetwarzanie wartości, która znajdowała się przed tym krokiem wstępnego przetwarzania. Należy pamiętać, że te błędy usług zewnętrznych są zgłaszane użytkownikowi bez zmian, bez dodawania informacji o kroku wstępnego przetwarzania. W przypadku niepowodzenia analizy nieprawidłowego kodu XML nie zostanie zgłoszony żaden błąd. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
Pasuje do wyrażenia regularnego	Określ wyrażenie regularne, do którego wartość musi pasować. Po zaznaczeniu pola wyboru *Niestandardowe działanie w przypadku błędu* można określić niestandardowe opcje obsługi błędów: odrzucić wartość, ustawić określoną wartość lub ustawić określony komunikat o błędzie.
Ograniczanie	
Odrzuć niezmienione z sygnałem kontrolnym	Odrzuć wartość, jeśli nie zmieniła się w określonym czasie (w sekundach). Do określenia liczby sekund można używać dodatnich wartości całkowitych (minimum - 1 sekunda). W tym polu można używać przyrostków czasu (np. `30s`, `1m`, `2h`, `1d`). W tym polu można używać makr użytkownika i makr wykrywania niskopoziomowego. Dla pozycji wykrywania można określić tylko jedną opcję ograniczania. Na przykład `1m`. Jeśli identyczny tekst zostanie przekazany do tej reguły dwukrotnie w ciągu 60 sekund, zostanie odrzucony. *Uwaga*: Zmiana prototypów pozycji nie resetuje ograniczania. Ograniczanie jest resetowane tylko po zmianie kroków wstępnego przetwarzania.
Prometheus	
Prometheus do JSON	Konwertuj wymagane metryki Prometheus do formatu JSON. Więcej informacji można znaleźć w sekcji Sprawdzanie Prometheus.

Należy pamiętać, że jeśli reguła wykrywania została zastosowana do hosta za pośrednictwem szablonu, zawartość tej karty jest tylko do odczytu.

Niestandardowe makra

Karta **LLD macros** umożliwia określenie niestandardowych makr LLD, gdy zwrócony JSON nie zawiera jeszcze kluczy w formacie `{#MACRO}`.

Na przykład klucz pozycji agenta `vfs.fs.get` zwraca dane JSON systemu plików bez żadnych predefiniowanych makr LLD.
Tutaj definiujesz makra ręcznie i mapujesz je na wartości w JSON za pomocą wyrażeń JSONPath:



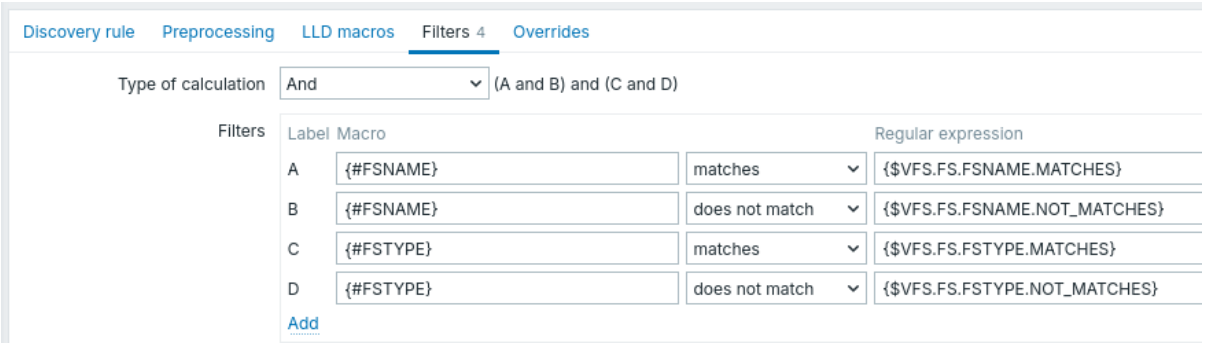
Parameter	Description
LLD macro	Nazwa makra LLD w formacie `{#MACRO}`.
JSONPath	Wyrażenie określające, jaką wartość należy wyodrębnić z każdej wykrytej encji w tablicy JSON. Na przykład `$.foo` wyodrębnia `"bar"` i `"baz"` z `[{"foo":"bar"}, {"foo":"baz"}]`. JSONPath obsługuje zarówno notację kropkową, jak i nawiasową. Notacji nawiasowej należy używać dla kluczy zawierających znaki specjalne lub Unicode, np. `$['unicode + special chars #1']['unicode + special chars #2']`.

Reguła wykrywania musi zwracać tablicę JSON w korzeniu (`$.`).
Ze względu na zgodność wsteczną akceptowany jest również obiekt JSON zawierający pojedynczą tablicę `"data"`. W takim przypadku tablica `"data"` jest automatycznie rozwijana:

```
{"data":[{"foo":"bar"}, {"foo":"baz"}]}
```

Filtr

Filtr może służyć do generowania rzeczywistych pozycji, wyzwalaczy i wykresów tylko dla encji spełniających określone kryteria. Karta **Filtry** zawiera definicje filtrów reguły wykrywania, które umożliwiają filtrowanie wartości wykrywania:



Parametr	Opis
Typ obliczeń	Dostępne są następujące opcje obliczania filtrów: **And** - wszystkie filtry muszą zostać spełnione; **Or** - wystarczy spełnienie jednego filtra; **And/Or** - używa *And* dla różnych nazw makr oraz *Or* dla tej samej nazwy makra; **Wyrażenie niestandardowe** - umożliwia zdefiniowanie niestandardowego obliczania filtrów. Formuła musi zawierać wszystkie filtry z listy. Ograniczenie do 255 znaków.

Parametr	Opis
Filtry	Dostępne są następujące operatory warunków filtrowania: *pasuje*, *nie pasuje*, *istnieje*, *nie istnieje*. Operatory *pasuje* i *nie pasuje* wymagają wyrażenia regularnego zgodnego z Perlem (PCRE). W polu *Wyrażenie regularne* można wprowadzić wyrażenie regularne lub odwołać się do globalnego wyrażenia regularnego. Operatory *istnieje* i *nie istnieje* umożliwiają filtrowanie encji na podstawie obecności lub braku określonego makra LLD w odpowiedzi. Należy pamiętać, że jeśli w odpowiedzi brakuje makra z filtra, wykryta encja zostanie zignorowana, chyba że dla tego makra określono warunek „nie istnieje”. Jeśli brak makra wpływa na wynik wyrażenia, zostanie wyświetlone ostrzeżenie. Na przykład, jeśli w wyrażeniu `{#B}` brakuje: `{#A}` pasuje do 1 i `{#B}` pasuje do 2 - zostanie wyświetlone ostrzeżenie `{#A}` pasuje do 1 lub `{#B}` pasuje do 2 - ostrzeżenie nie zostanie wyświetlone.

Na przykład, jeśli interesują Cię tylko systemy plików C:, D: i E:, możesz umieścić `{#FSNAME}` w polu *Makro*, a `^C|^D|^E` w polu *Wyrażenie regularne*.
 Filtrowanie jest również możliwe według typów systemów plików za pomocą makra `{#FSTYPE}` (np. `^ext|^reiserfs`) oraz według typów dysków za pomocą makra `{#FSDRIVETYPE}` (np. `fixed`) (obsługiwane tylko przez Windows agent).

Aby przetestować wyrażenie regularne, można użyć polecenia `grep -E`, na przykład:

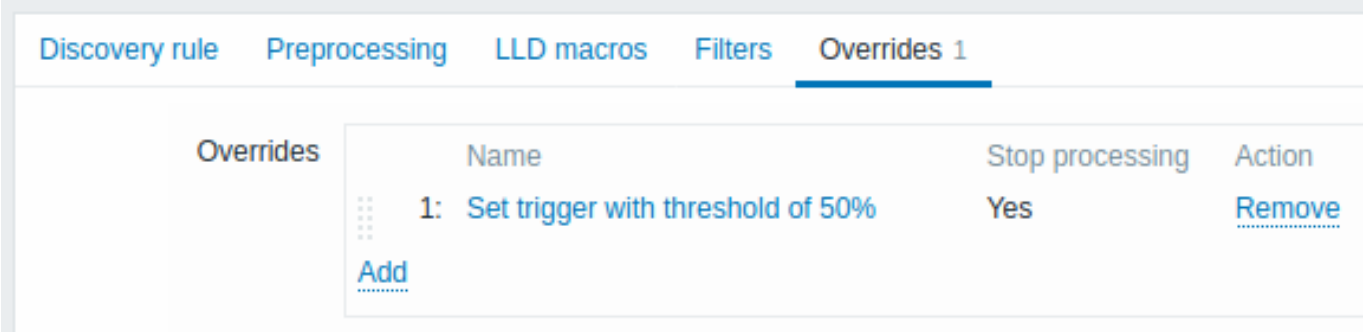
```
for f in ext2 nfs reiserfs smbfs;
do echo $f | grep -E '^ext|^reiserfs' || echo "SKIP: $f";
done
```

Warning:
Błąd lub literówka w wyrażeniu regularnym używanym w regule LLD (na przykład nieprawidłowe wyrażenie regularne „Systemy plików do wykrycia”) może spowodować usunięcie tysięcy elementów konfiguracji, wartości historycznych i zdarzeń dla wielu hostów.

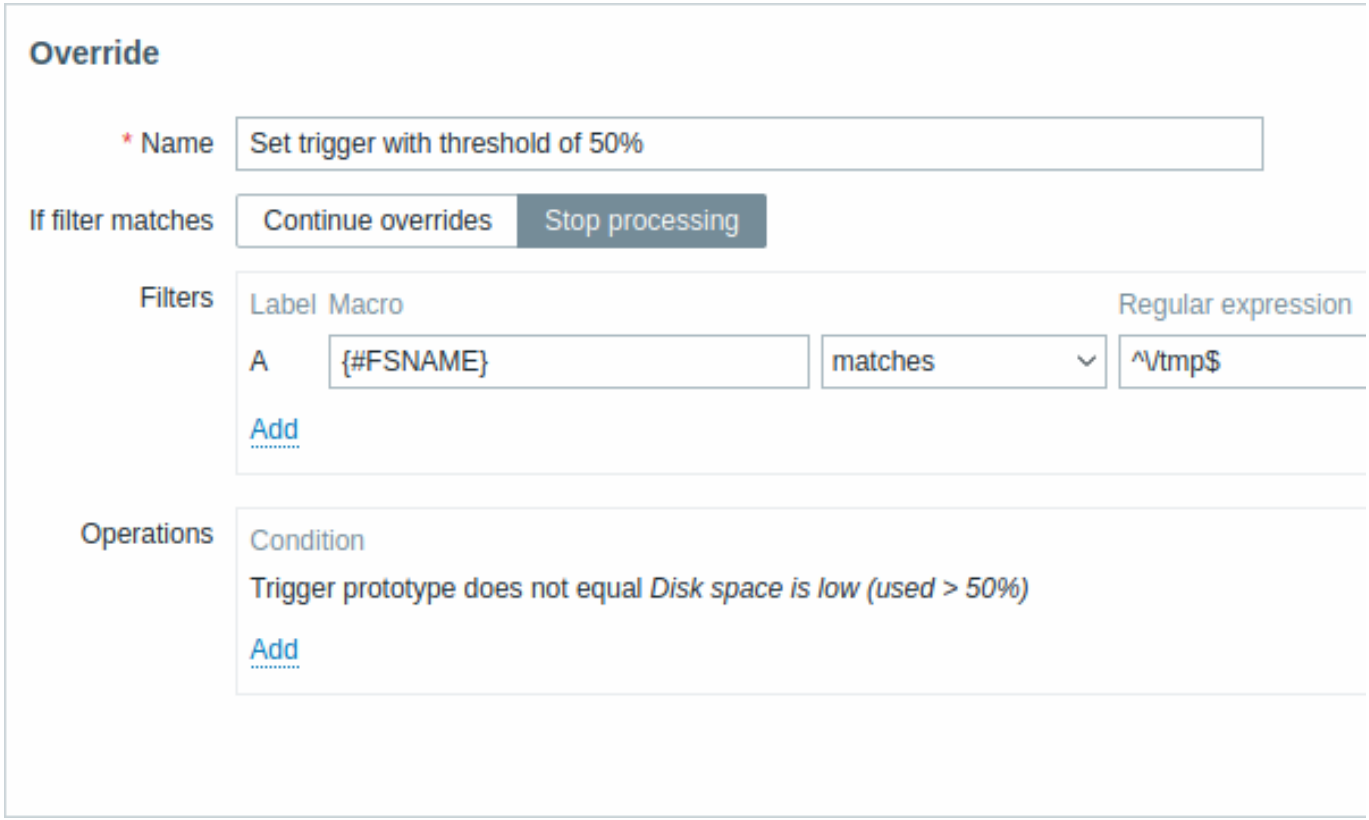
Attention:
Baza danych Zabbix w MySQL musi zostać utworzona z uwzględnieniem wielkości liter, aby nazwy systemów plików różniące się tylko wielkością liter były wykrywane prawidłowo.

Nadpisywanie

Karta **Nadpisywania** umożliwia skonfigurowanie reguł modyfikowania listy prototypów pozycji, wyzwalaczy, wykresów, hostów i reguł wykrywania lub ich atrybutów dla wykrytych obiektów spełniających określone kryteria.



Nadpisywania (jeśli istnieją) są wyświetlane na liście, której kolejność można zmieniać metodą przeciągania i upuszczania, oraz wykonywane w kolejności, w której zostały zdefiniowane. Aby skonfigurować szczegóły nowego nadpisywania, kliknij Add w bloku *Nadpisywania*. Aby edytować istniejące nadpisywanie, kliknij jego nazwę. Otworzy się wyskakujące okno umożliwiające edycję szczegółów reguły nadpisywania.

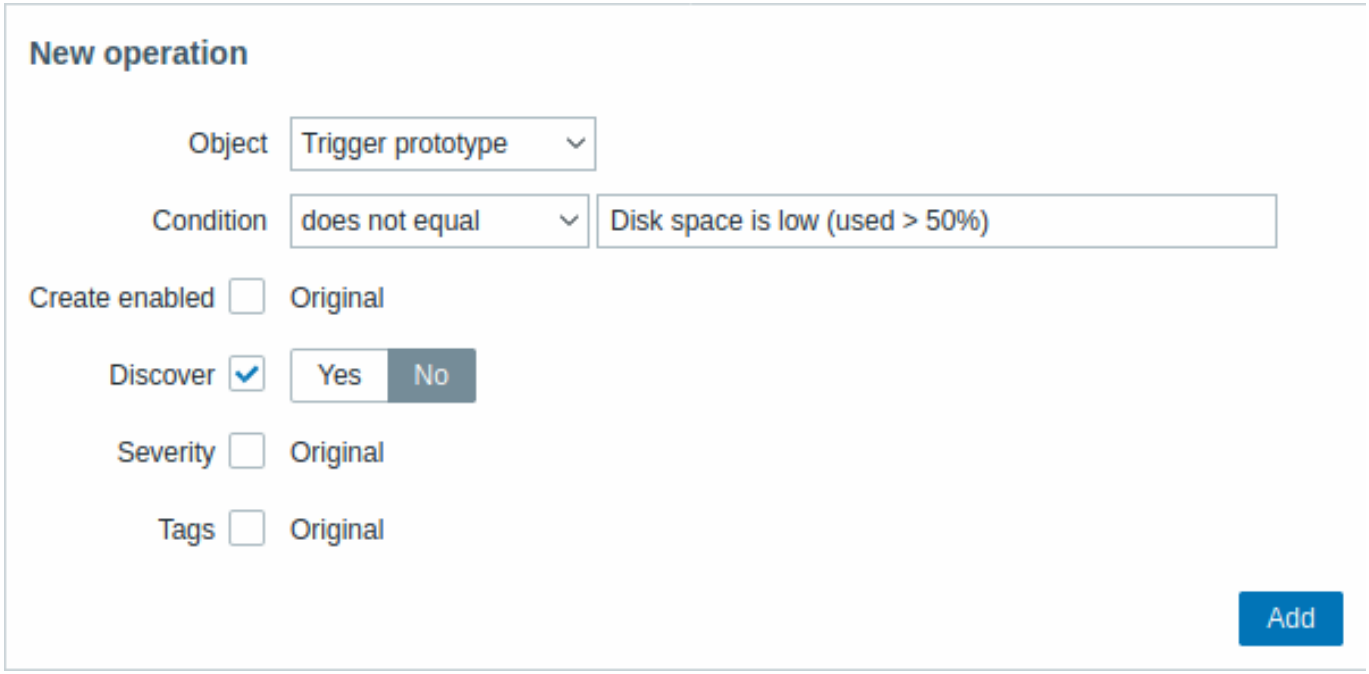


Wszystkie wymagane parametry są oznaczone czerwonymi gwiazdkami.

Parametr	Opis
Nazwa	Unikalna (w ramach reguły LLD) nazwa nadpisywania.
Jeśli filtr pasuje	Określa, czy należy przetwarzać kolejne nadpisywania, gdy warunki filtra są spełnione: **Kontynuuj nadpisywania** - kolejne nadpisywania będą przetwarzane. **Zatrzymaj przetwarzanie** - zostaną wykonane operacje z poprzednich nadpisań (jeśli istnieją) oraz tego nadpisywania, a kolejne nadpisywania zostaną pominięte dla pasujących wierszy LLD.
Filtry	Określa, do których wykrytych jednostek należy zastosować nadpisywanie. Filtry nadpisywania są przetwarzane po filtrach reguły wykrywania i mają taką samą funkcjonalność.
Operacje	Operacje nadpisywania są wyświetlane wraz z następującymi szczegółami: **Warunek** - typ obiektu i warunek, który musi być spełniony dla nazwy obiektu; na przykład: prototyp wyzwalacza nie jest równy *Mało miejsca na dysku (użyto > 50%)*. **Działania** - wyświetlane są odnośniki umożliwiające edycję i usunięcie operacji.

Konfigurowanie operacji

Aby skonfigurować szczegóły nowej operacji, kliknij Add w bloku Operacje. Aby edytować istniejącą operację, kliknij Edit obok operacji. Otworzy się wyskakujące okno umożliwiające edycję szczegółów operacji.



Parametr	Opis
Obiekt	Dostępnych jest pięć typów obiektów: Prototyp pozycji Prototyp wyzwalacza Prototyp wykresu Prototyp hosta Prototyp reguły wykrywania
Warunek	Umożliwia filtrowanie jednostek, do których należy zastosować operację.
Operator	Obsługiwane operatory: **jest równy** - zastosuj do tego prototypu **nie jest równy** - zastosuj do wszystkich prototypów oprócz tego **zawiera** - zastosuj, jeśli nazwa prototypu zawiera ten ciąg znaków **nie zawiera** - zastosuj, jeśli nazwa prototypu nie zawiera tego ciągu znaków **pasuje** - zastosuj, jeśli nazwa prototypu pasuje do wyrażenia regularnego **nie pasuje** - zastosuj, jeśli nazwa prototypu nie pasuje do wyrażenia regularnego
Wzorzec	Wyrażenie regularne lub ciąg znaków do wyszukania.
Obiekt: *Prototyp pozycji*	
Utwórz jako włączoną	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu pozycji: *Tak* - pozycja zostanie dodana w stanie włączonym. *Nie* - pozycja zostanie dodana do wykrytej jednostki, ale w stanie wyłączonym.
Wykrywaj	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu pozycji: *Tak* - pozycja zostanie dodana. *Nie* - pozycja nie zostanie dodana.
Interwał aktualizacji	Po zaznaczeniu pola wyboru pojawią się dwie opcje umożliwiające ustawienie innego interwału dla pozycji: *Opóźnienie* - interwał aktualizacji pozycji. Obsługiwane są makra użytkownika, makra LLD oraz przyrostki czasu (np. 30s, 1m, 2h, 1d). Należy ustawić wartość 0, jeśli używany jest *Interwał niestandardowy*. *Interwał niestandardowy* - kliknij Add, aby określić elastyczne/harmonogramowe interwały. Szczegółowe informacje zawiera sekcja Interwały niestandardowe.

Parametr	Opis
Historia	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające ustawienie innych okresów przechowywania historii dla pozycji: *Nie przechowuj* - po wybraniu historia nie będzie przechowywana. *Przechowuj przez* - po wybraniu po prawej stronie pojawi się pole do określenia okresu przechowywania. Obsługiwane są makra użytkownika i makra LLD.
Trendy	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające ustawienie innych okresów przechowywania trendów dla pozycji: *Nie przechowuj* - po wybraniu trendy nie będą przechowywane. *Przechowuj przez* - po wybraniu po prawej stronie pojawi się pole do określenia okresu przechowywania. Obsługiwane są makra użytkownika i makra LLD.
Znaczniki	Po zaznaczeniu pola wyboru pojawi się nowy blok umożliwiający określenie par znacznik-wartość. Obsługiwane są makra użytkownika i makra LLD. Te znaczniki zostaną dodane do znaczników określonych w prototypie pozycji, nawet jeśli nazwy znaczników są takie same.
Obiekt: *Prototyp wyzwalacza*	
Utwórz jako włączony	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu wyzwalacza: *Tak* - wyzwalacz zostanie dodany w stanie włączonym. *Nie* - wyzwalacz zostanie dodany do wykrytej jednostki, ale w stanie wyłączonym.
Wykrywaj	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu wyzwalacza: *Tak* - wyzwalacz zostanie dodany. *Nie* - wyzwalacz nie zostanie dodany.
Ważność	Po zaznaczeniu pola wyboru pojawią się przyciski ważności wyzwalacza umożliwiające zmianę jego ważności.
Znaczniki	Po zaznaczeniu pola wyboru pojawi się nowy blok umożliwiający określenie par znacznik-wartość. Obsługiwane są makra użytkownika i makra LLD. Te znaczniki zostaną dodane do znaczników określonych w prototypie wyzwalacza, nawet jeśli nazwy znaczników są takie same.
Obiekt: *Prototyp wykresu*	
Wykrywaj	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu wykresu: *Tak* - wykres zostanie dodany. *Nie* - wykres nie zostanie dodany.
Obiekt: *Prototyp hosta*	
Utwórz jako włączony	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu hosta: *Tak* - host zostanie utworzony w stanie włączonym. *Nie* - host zostanie utworzony w stanie wyłączonym.
Wykrywaj	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu hosta: *Tak* - host zostanie wykryty. *Nie* - host nie zostanie wykryty.
Połącz szablony	Po zaznaczeniu pola wyboru pojawi się pole umożliwiające określenie szablonów. Zacznij wpisywać nazwę szablonu lub kliknij *Wybierz* obok pola i wybierz szablony z listy w wyskakującym oknie. Szablony z tego nadpisywania zostaną dodane do wszystkich szablonów już połączonych z prototypem hosta.

Parametr		Opis
	Znaczniki	Po zaznaczeniu pola wyboru pojawi się nowy blok umożliwiający określenie par znacznik-wartość. Obsługiwane są makra użytkownika i makra LLD. Te znaczniki zostaną dodane do znaczników określonych w prototypie hosta, nawet jeśli nazwy znaczników są takie same.
	Inwentaryzacja hosta	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające wybranie innego trybu inwentaryzacji dla prototypu hosta: *Wyłączona* - nie wypełniaj inwentaryzacji hosta *Ręczna* - wprowadź szczegóły ręcznie *Automatyczna* - automatycznie wypełniaj dane inwentaryzacji hosta na podstawie zebranych metryk.
Obiekt: *Prototyp reguły wykrywania*		
	Utwórz jako włączoną	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu reguły wykrywania: *Tak* - reguła wykrywania zostanie dodana w stanie włączonym. *Nie* - reguła wykrywania zostanie dodana w stanie wyłączonym.
	Wykrywaj	Po zaznaczeniu pola wyboru pojawią się przyciski umożliwiające nadpisanie oryginalnych ustawień prototypu reguły wykrywania: *Tak* - reguła wykrywania zostanie dodana. *Nie* - reguła wykrywania nie zostanie dodana.
	Interwał aktualizacji	Po zaznaczeniu pola wyboru pojawią się dwie opcje umożliwiające ustawienie innego interwału dla reguły wykrywania: *Opóźnienie* - interwał aktualizacji reguły. Obsługiwane są makra użytkownika, makra LLD oraz przyrostki czasu (np. 30s, 1m, 2h, 1d). Należy ustawić wartość 0, jeśli używany jest *Interwał niestandardowy*. *Interwał niestandardowy* - kliknij Add, aby określić elastyczne/harmonogramowe interwały. Szczegółowe informacje zawiera sekcja Interwały niestandardowe.

Przyciski formularza

Przyciski u dołu formularza umożliwiają wykonanie kilku operacji.

Add	Dodaj regułę wykrywania. Ten przycisk jest dostępny tylko dla nowych reguł wykrywania.
Update	Zaktualizuj właściwości reguły wykrywania. Ten przycisk jest dostępny tylko dla istniejących reguł wykrywania.
Clone	Utwórz kolejną regułę wykrywania na podstawie właściwości bieżącej reguły wykrywania.
Execute now	Wykonaj natychmiast wykrywanie na podstawie reguły wykrywania. Reguła wykrywania musi już istnieć. Zobacz więcej szczegółów. *Uwaga*: podczas natychmiastowego wykonywania wykrywania pamięć podręczna konfiguracji nie jest aktualizowana, dlatego wynik nie będzie uwzględniał bardzo niedawnych zmian w konfiguracji reguły wykrywania.
Test	Przetestuj konfigurację reguły wykrywania. Użyj tego przycisku, aby zweryfikować ustawienia konfiguracji (takie jak łączność i poprawność parametrów) bez trwałego zastosowania zmian.
Delete	Usuń regułę wykrywania.
Cancel	Anuluj edytowanie właściwości reguły wykrywania.

Odkryte encje Poniższe zrzuty ekranu ilustrują, jak odkryte pozycje, wyzwalacze i wykresy wyglądają w konfiguracji hosta. Odkryte encje są poprzedzone pomarańczowym odnośnikiem do reguły wykrywania, z której pochodzą.

Items

All hosts / Zabbix server Enabled ZBX SNMP IPMI JMX Items 140 Triggers 77 Graphs 14 Discovery rules 6 Web scenarios

Name ▲	Triggers	Key	Interval
Mounted filesystem discovery: Get filesystems: FS [/]: Get data		vfs.fs.dependent[/,data]	
Mounted filesystem discovery: FS [/]: Get data: FS [/]: Inodes: Free, in %	Triggers 2	vfs.fs.dependent.inode[/,pfree]	
Mounted filesystem discovery: FS [/]: Get data: FS [/]: Option: Read-only	Triggers 1	vfs.fs.dependent[/,readonly]	
Mounted filesystem discovery: FS [/]: Get data: FS [/]: Space: Available		vfs.fs.dependent.size[/,free]	
Mounted filesystem discovery: FS [/]: Get data: FS [/]: Space: Total		vfs.fs.dependent.size[/,total]	
Mounted filesystem discovery: FS [/]: Get data: FS [/]: Space: Used		vfs.fs.dependent.size[/,used]	
Mounted filesystem discovery: FS [/]: Get data: FS [/]: Space: Used, in %	Triggers 2	vfs.fs.dependent.size[/,pused]	

Należy pamiętać, że odkryte encje nie zostaną utworzone, jeśli istnieją już encje o tych samych kryteriach unikalności, na przykład pozycja z takim samym kluczem lub wykres o tej samej nazwie. W takim przypadku w frontend jest wyświetlany komunikat o błędzie informujący, że reguła wykrywania niskiego poziomu nie mogła utworzyć niektórych encji. Sama reguła wykrywania nie stanie się jednak nieobsługiwana tylko dlatego, że nie udało się utworzyć jakiejś encji i trzeba było ją pominąć. Reguła wykrywania będzie nadal tworzyć/aktualizować inne encje.

Jeśli odkryta encja (host, system plików, interfejs itp.) przestanie być wykrywana (lub nie przejdzie już filtra), encje utworzone na jej podstawie mogą zostać automatycznie wyłączone, a ostatecznie usunięte.

Utracone zasoby mogą być automatycznie wyłączane na podstawie wartości parametru *Disable lost resources*. Dotyczy to utraconych hostów, pozycji i wyzwalaczy.

Utracone zasoby mogą być automatycznie usuwane na podstawie wartości parametru *Delete lost resources*. Dotyczy to utraconych hostów, grup hostów, pozycji, wyzwalaczy i wykresów.

Gdy odkryte encje przechodzą do stanu „Not discovered anymore", na liście encji wyświetlany jest wskaźnik czasu życia. Po najechaniu na niego kursorem myszy zostanie wyświetlony komunikat zawierający szczegóły statusu.



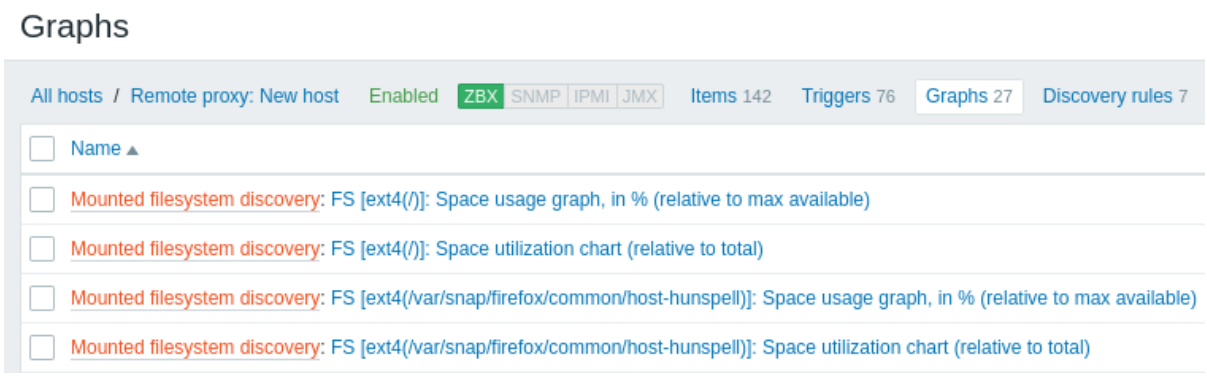
Jeśli encje zostały oznaczone do usunięcia, ale nie zostały usunięte w oczekiwanym czasie (wyłączona reguła wykrywania lub host pozycji), zostaną usunięte przy następnym przetwarzaniu reguły wykrywania.

Encje zawierające inne encje oznaczone do usunięcia nie będą aktualizowane, jeśli zostaną zmienione na poziomie reguły wykrywania. Na przykład wyzwalacze oparte na LLD nie będą aktualizowane, jeśli zawierają pozycje oznaczone do usunięcia.

Triggers

All hosts / Zabbix server Enabled ZBX SNMP IPMI JMX Items 140 Triggers 77 Graphs 14 Discovery rules 6

Severity	Value	Name ▲	Operational data
Warning	OK	Mounted filesystem discovery: Linux: FS [/]: Running out of free inodes **Depends on:** Zabbix server: Linux: FS [/]: Running out of free inodes	Free inodes: {ITEM.LASTVALUE1}
Warning	OK	Mounted filesystem discovery: Linux: FS [/]: Space is low **Depends on:** Zabbix server: Linux: FS [/]: Space is critically low	Space used: {{ITEM.LASTVALUE1}.fmtnum(1)}%



Inne typy wykrywania Więcej szczegółów oraz instrukcje dotyczące innych gotowych typów wykrywania są dostępne w następujących sekcjach:

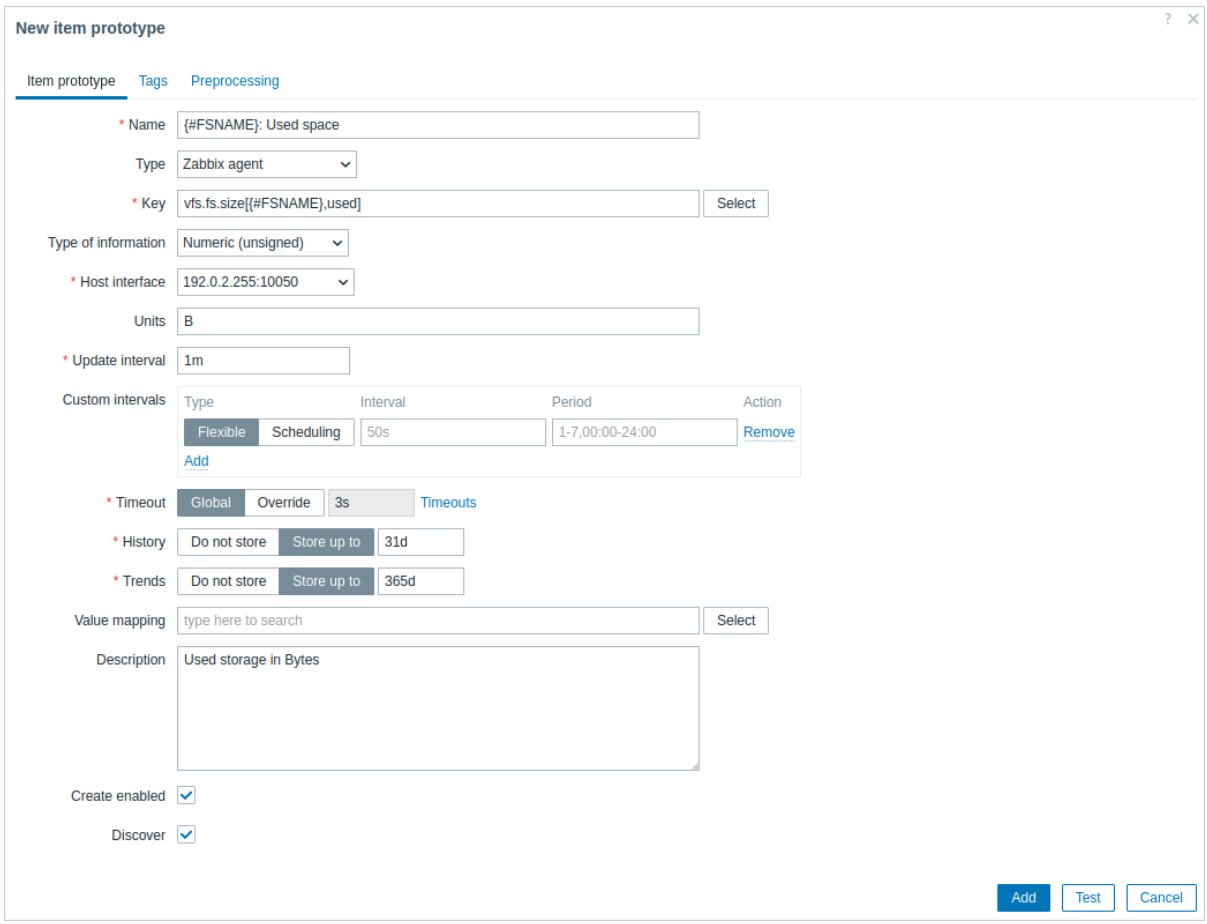
- wykrywanie interfejsów sieciowych
- wykrywanie procesorów i rdzeni CPU
- wykrywanie OID-ów SNMP
- wykrywanie obiektów JMX;
- wykrywanie przy użyciu zapytań SQL ODBC
- wykrywanie usług Windows
- wykrywanie interfejsów hosta w Zabbix

Więcej szczegółów na temat formatu JSON dla pozycji wykrywania oraz przykład implementacji własnego mechanizmu wykrywania systemu plików jako skryptu Perl można znaleźć w sekcji tworzenie niestandardowych reguł LLD.

Prototypy pozycji

Po utworzeniu reguły przejdź do pozycji tej reguły i kliknij *Utwórz prototyp pozycji*, aby utworzyć prototyp pozycji.

Zwróć uwagę, że makro `{#FSNAME}` jest używane w miejscach, w których wymagana jest nazwa systemu plików. Użycie makra wykrywania niskopoziomowego w kluczu pozycji jest obowiązkowe, aby zapewnić prawidłowe przetwarzanie wykrywania. Podczas przetwarzania reguły wykrywania to makro zostanie zastąpione wykrytym systemem plików.



Note:
Jeśli parametry klucza pozycji nie są używane, umieść makro LLD wewnątrz nawiasów klucza pozycji `[...]` (na przykład `v_[{#MACRO}]`).

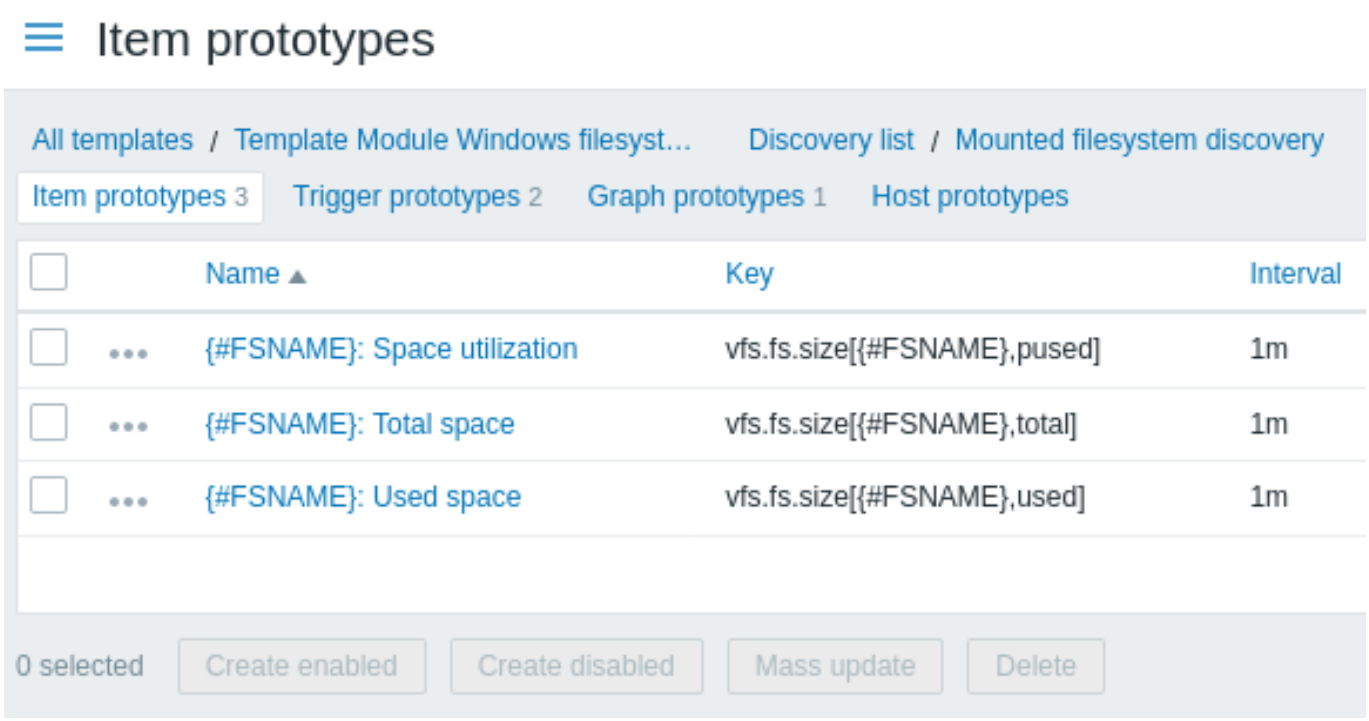
Makra wykrywania niskopoziomowego macros oraz makra użytkownika macros są obsługiwane w konfiguracji prototypu pozycji oraz w parametrach wstępnego przetwarzania wartości pozycji. Należy pamiętać, że w przypadku użycia w interwałach aktualizacji pojedyncze makro musi wypełniać całe pole. Wiele makr w jednym polu lub makra połączone z tekstem nie są obsługiwane.

Note:
Kontekstowe zmienianie znaków specjalnych w makrach wykrywania niskopoziomowego jest wykonywane w celu zapewnienia bezpiecznego użycia w parametrach wstępnego przetwarzania wyrażeń regularnych i XPath.

Atrybuty charakterystyczne dla prototypów pozycji:

Parametr	Opis
Utwórz jako włączone	Jeśli opcja jest zaznaczona, pozycja zostanie dodana w stanie włączonym. Jeśli opcja nie jest zaznaczona, pozycja zostanie dodana do wykrytej jednostki w stanie wyłączonym.
Wykrywaj	Jeśli opcja jest zaznaczona (domyślnie), pozycja zostanie dodana do wykrytej jednostki. Jeśli opcja nie jest zaznaczona, pozycja nie zostanie dodana do wykrytej jednostki, chyba że to ustawienie zostanie nadpisane w regule wykrywania.

Możemy utworzyć kilka prototypów pozycji dla każdej interesującej nas metryki systemu plików:



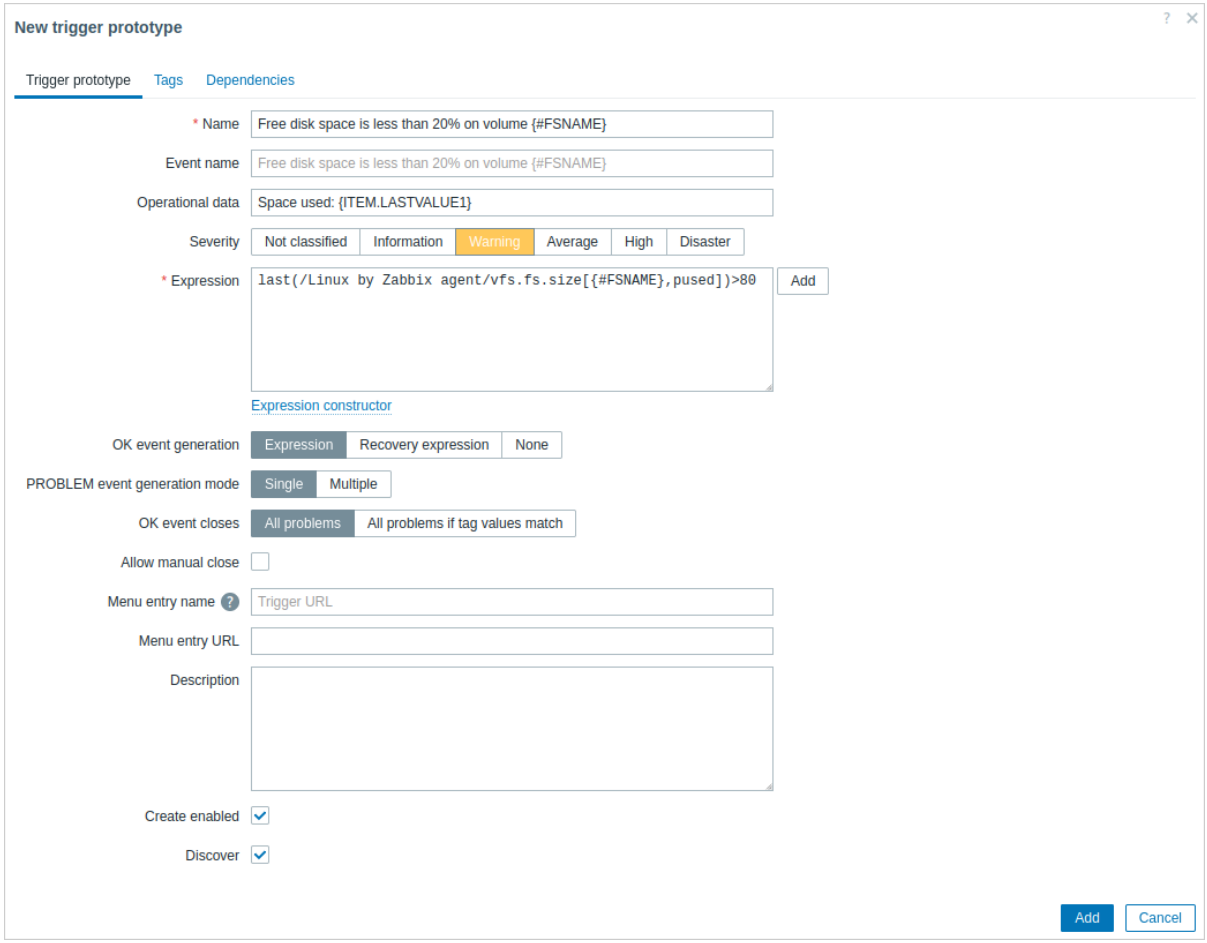
Kliknij ikonę trzech kropek, aby otworzyć menu konkretnego prototypu pozycji z następującymi opcjami:

- *Utwórz prototyp wyzwalacza* - utwórz prototyp wyzwalacza na podstawie tego prototypu pozycji
- *Prototypy wyzwalaczy* - kliknij, aby wyświetlić listę zawierającą odnośniki do skonfigurowanych już prototypów wyzwalaczy tego prototypu pozycji
- *Utwórz pozycję zależną* - utwórz pozycję zależną dla tego prototypu pozycji

Opcja *Aktualizacja zbiorcza* jest dostępna, jeśli chcesz jednocześnie zaktualizować właściwości kilku prototypów pozycji.

Prototypy wyzwalaczy

Tworzymy prototypy wyzwalaczy w podobny sposób jak prototypy pozycji:

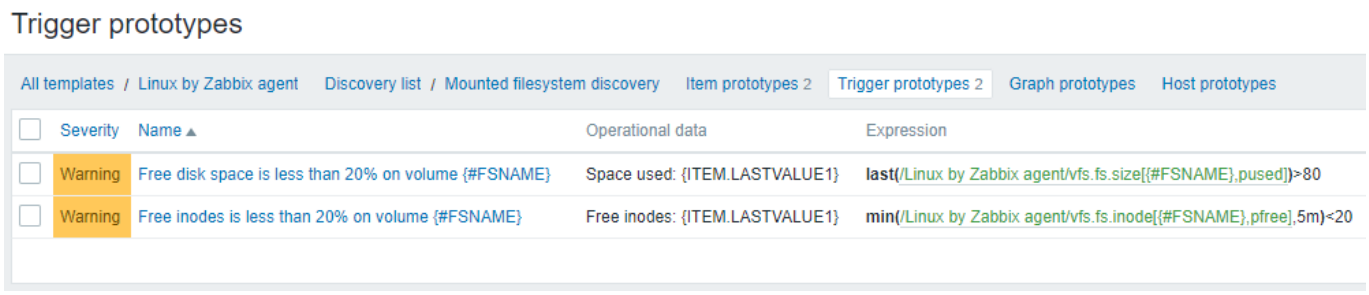


Atrybuty specyficzne dla prototypów wyzwalaczy:

Parameter	Description
Create enabled	Jeśli zaznaczone, wyzwalacz zostanie dodany w stanie włączonym. Jeśli odznaczone, wyzwalacz zostanie dodany do wykrytej encji, ale w stanie wyłączonym.
Discover	Jeśli zaznaczone (domyślnie), wyzwalacz zostanie dodany do wykrytej encji. Jeśli odznaczone, wyzwalacz nie zostanie dodany do wykrytej encji, chyba że to ustawienie zostanie zastąpione w regule wykrywania.

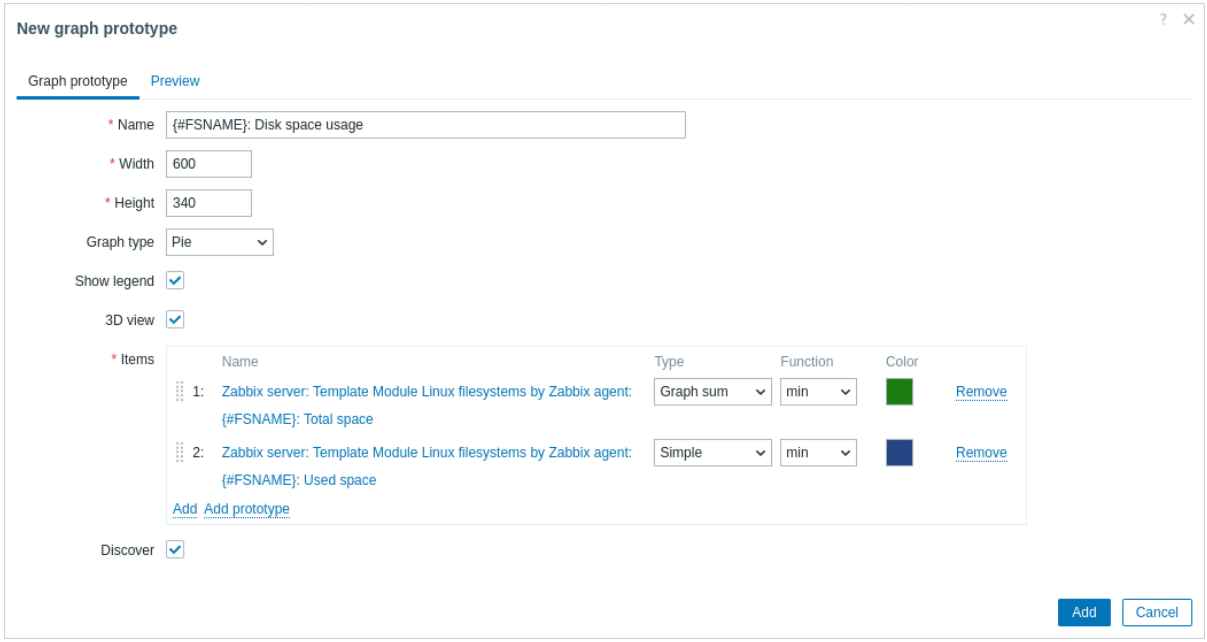
Gdy rzeczywiste wyzwalacze są tworzone na podstawie prototypów, może pojawić się potrzeba elastycznego określenia, jaka stała ('20' w naszym przykładzie) ma być użyta do porównania w wyrażeniu. Zobacz, jak makra użytkownika z kontekstem mogą być przydatne do osiągnięcia takiej elastyczności.

Możesz zdefiniować zależności między prototypami wyzwalaczy. Aby to zrobić, przejdź do karty *Dependencies*. Prototyp wyzwalacza może zależeć od innego prototypu wyzwalacza z tej samej reguły niskopoziomowego wykrywania (LLD) lub od zwykłego wyzwalacza. Prototyp wyzwalacza nie może zależeć od prototypu wyzwalacza z innej reguły LLD ani od wyzwalacza utworzonego na podstawie prototypu wyzwalacza. Prototyp wyzwalacza hosta nie może zależeć od wyzwalacza z szablonu.



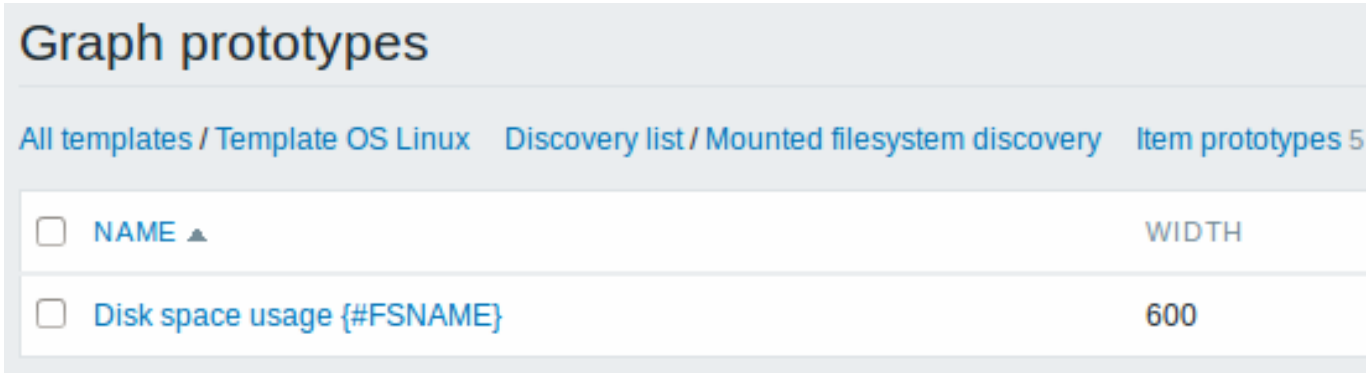
Prototypy wykresów

Możemy również tworzyć prototypy wykresów:

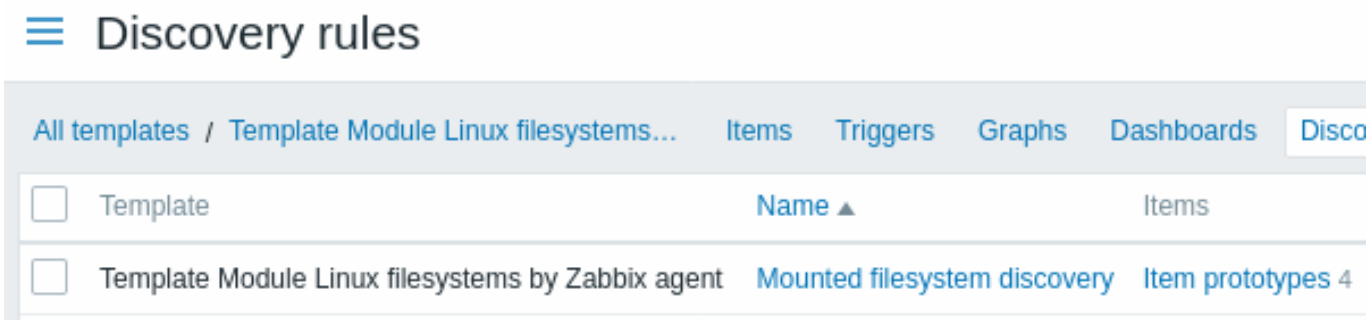


Atrybuty specyficzne dla prototypów wykresów:

Parameter	Description
Discover	Jeśli zaznaczone (domyślnie), wykres zostanie dodany do wykrytej encji. Jeśli odznaczone, wykres nie zostanie dodany do wykrytej encji, chyba że to ustawienie zostanie zastąpione w regule wykrywania.



Na koniec utworzyliśmy regułę wykrywania, która wygląda jak poniżej. Ma pięć prototypów pozycji, dwa prototypy wyzwalaczy i jeden prototyp wykresu.



Prototypy hostów

Prototypy hostów są wzorcami do tworzenia hostów za pomocą reguł odkrywania niskiego poziomu. Zanim zostaną odkryte jako hosty, te prototypy nie mogą mieć pozycji ani wyzwalaczy, z wyjątkiem tych podłączonych z szablonów.

Konfiguracja

Prototypy hostów są konfigurowane w ramach reguł wykrywania niskiego poziomu.

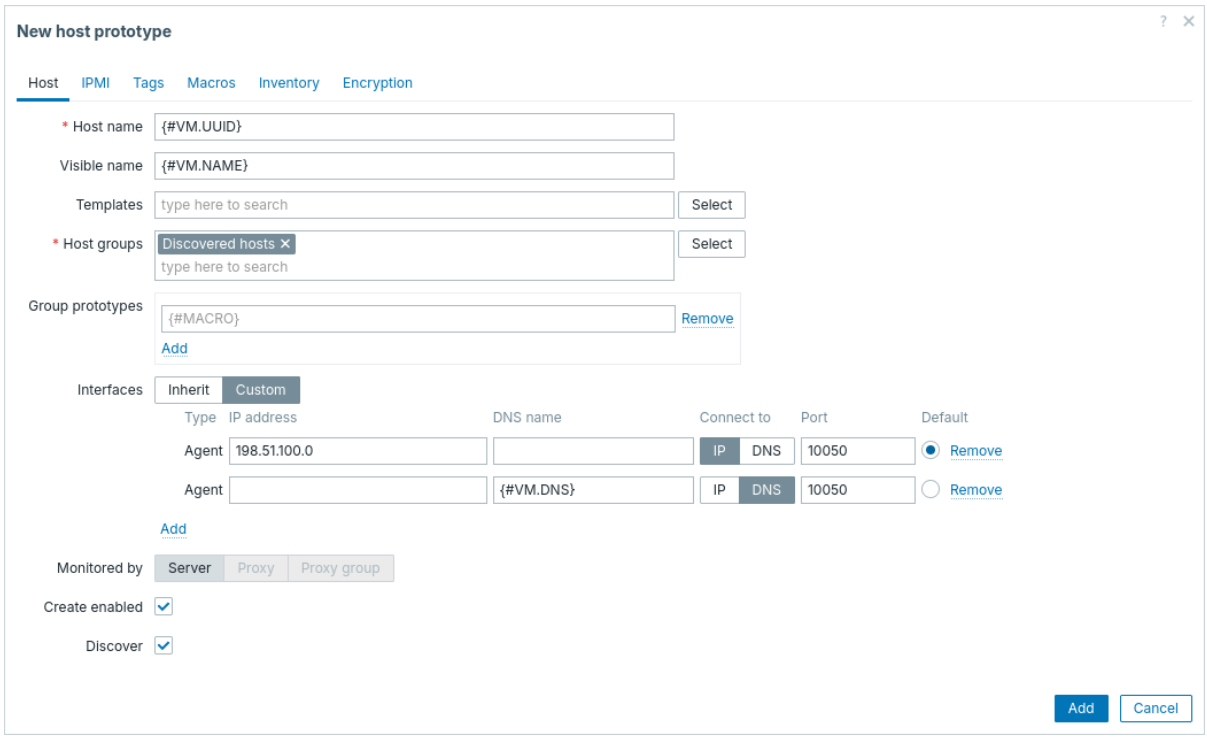
Aby utworzyć prototyp hosta:

1. Przejdź do *Zbieranie danych > Hosty*.

2. Kliknij *Discovery* dla wymaganego hosta, aby przejść do listy reguł wykrywania niskiego poziomu skonfigurowanych dla tego hosta.

3. Kliknij *Prototypy hostów* dla wymaganej reguły wykrywania.

4. Kliknij przycisk *Utwórz prototyp hosta* w prawym górnym rogu i wypełnij formularz konfiguracji.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Host name	Ten parametr musi zawierać co najmniej jeden makro wykrywania niskiego poziomu, aby zapewnić unikalne nazwy hostów dla tworzonych hostów.
Visible name	Obsługiwane są makra wykrywania niskiego poziomu.
Group prototypes	Umożliwia określanie prototypów grup hostów za pomocą makr wykrywania niskiego poziomu. Na podstawie określonych prototypów grup zostaną wykryte, utworzone i powiązane z utworzonymi hostami grupy hostów; wykryte grupy, które zostały już utworzone przez inne reguły wykrywania niskiego poziomu, również zostaną powiązane z utworzonymi hostami. Jednak wykryte grupy hostów, które odpowiadają grupom hostów utworzonym ręcznie, nie zostaną powiązane z utworzonymi hostami.
Interfaces	Określa, czy wykryte hosty dziedziczą adres IP z hosta, do którego należy reguła wykrywania (domyślnie), czy otrzymują niestandardowe interfejsy. Obsługiwane są makra wykrywania niskiego poziomu oraz makra użytkownika.
Monitored by	To pole jest automatycznie synchronizowane z odpowiednim polem w konfiguracji hosta.
Create enabled	Ustawia status wykrytych hostów; jeśli opcja nie jest zaznaczona, hosty zostaną utworzone jako wyłączone.
Discover	Określa, czy hosty będą tworzone na podstawie prototypu hosta; jeśli opcja nie jest zaznaczona, hosty nie będą tworzone na podstawie prototypu hosta (chyba że to ustawienie zostanie zastąpione w regule wykrywania niskiego poziomu).

Note:
Makra wykrywania niskiego poziomu są również obsługiwane dla wartości tagów oraz wartości makr użytkownika prototypu hosta.
 Mapy wartości nie są obsługiwane dla prototypów hostów.

Przykład konfiguracji prototypu hosta można znaleźć w sekcji *Monitorowanie maszyn wirtualnych*.

Interfejsy hosta

Aby dodać niestandardowe interfejsy, przełącz selektor *Interfaces* z „Inherit” na „Custom”. Kliknij Add i wybierz typ interfejsu — Zabbix agent, SNMP, JMX, IPMI.

Note:
Jeśli wybrano *Custom*, ale nie ustawiono żadnych interfejsów, hosty zostaną utworzone bez interfejsów.
 Jeśli wybrano *Inherit* i prototyp hosta należy do szablonu, wszystkie wykryte hosty odziedziczą interfejs hosta od hosta, do którego podłączony jest szablon.

Jeśli określono wiele niestandardowych interfejsów, podstawowy interfejs można ustawić w kolumnie *Default*.

Przykład konfiguracji niestandardowego interfejsu hosta znajduje się w *przykładzie konfiguracji monitorowania VMware*.

Warning:
Host zostanie utworzony tylko wtedy, gdy interfejs hosta zawiera poprawne dane.

Wykryte hosty

Na liście hostów wykryte hosty są poprzedzone nazwą reguły wykrywania, która je utworzyła.

Wykryte hosty dziedziczą większość parametrów z prototypów hostów jako *tylko do odczytu*. Skonfigurować można tylko następujące parametry wykrytego hosta:

- *Szablony* - połączyć nowe szablony lub odłączyć dodane ręcznie. Szablonów odziedziczonych z prototypu hosta nie można odłączyć.
- *Opis* - dodać lub edytować opis hosta.
- *Status* - włączyć lub wyłączyć host.
- *Tagi* - dodać nowe tagi lub usunąć dodane ręcznie. Tagów odziedziczonych z prototypu hosta nie można usunąć. Należy pamiętać, że tagi nie mogą się powielać (tagi o tej samej nazwie i wartości). Jeśli do prototypu hosta zostanie dodany nowy tag i będzie on odpowiadał tagowi dodanemu ręcznie w wykrytym hoście, podczas wykrywania tag dodany ręcznie zostanie zastąpiony tagiem odziedziczonym.
- *Makra* - dodać nowe makra hosta lub usunąć dodane ręcznie, a także te odziedziczone z prototypu hosta; zmienić wartości makr i typy.

Note:
Wykryte hosty dziedziczą również makra użytkownika z hosta, na którym skonfigurowano regułę wykrywania. Te odziedziczone makra można usunąć, a ich wartości i typy można zmienić.

Wykryte hosty mogą również wykrywać inne hosty. Na przykład, jeśli prototyp hosta (używany do wykrywania hiperwizorów) ma podłączony szablon zawierający regułę wykrywania niskiego poziomu z własnym prototypem hosta (używanym do wykrywania maszyn wirtualnych), Zabbix wykryje zarówno hiperwizory, jak i ich maszyny wirtualne. Aby dodatkowo wykrywać kontenery na tych maszynach wirtualnych, można utworzyć reguły wykrywania niskiego poziomu lub podłączyć nowy szablon z prototypami hostów do wykrytych maszyn wirtualnych, albo wstępnie skonfigurować prototyp hosta maszyny wirtualnej przez podłączenie szablonu, który sam zawiera prototypy hostów.

Wykryte hosty można usuwać ręcznie. Należy jednak pamiętać, że zostaną wykryte ponownie, jeśli wykrywanie będzie dla nich włączone.

Hosty, które nie są już wykrywane, mogą zostać:

- automatycznie wyłączone (na podstawie wartości *Disable lost resources* reguły wykrywania)
- automatycznie usunięte (na podstawie wartości *Delete lost resources* reguły wykrywania).

Prototypy wykrywania

Przegląd

Prototypy wykrywania to **zagnieżdżone reguły wykrywania niskopoziomowego** w ramach „nadrzędnej” reguły wykrywania, umożliwiające wielopoziomowe wykrywanie obiektów wraz z ich własnymi pozycjami, wyzwalaczami itd. Na przykład można chcieć wykryć wszystkie instancje baz danych na serwerze baz danych, następnie wykryć obszary tabel dla każdej instancji, a potem tabele dla każdego obszaru tabel.

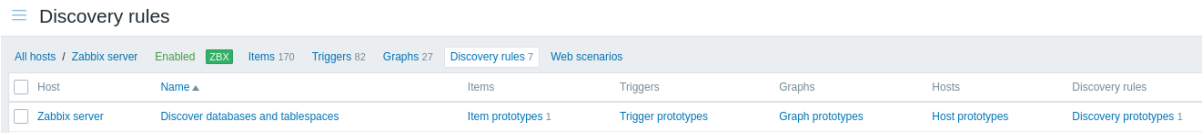
Prototypy wykrywania mają własne prototypy pozycji, wyzwalaczy, wykresów, hostów i wykrywania. Zagnieżdżony prototyp wykrywania będzie używać tej samej wartości JSON co reguła nadrzędna, jeśli określisz typ *Nested*.

Liczba poziomów zagnieżdżenia prototypów wykrywania jest nieograniczona.

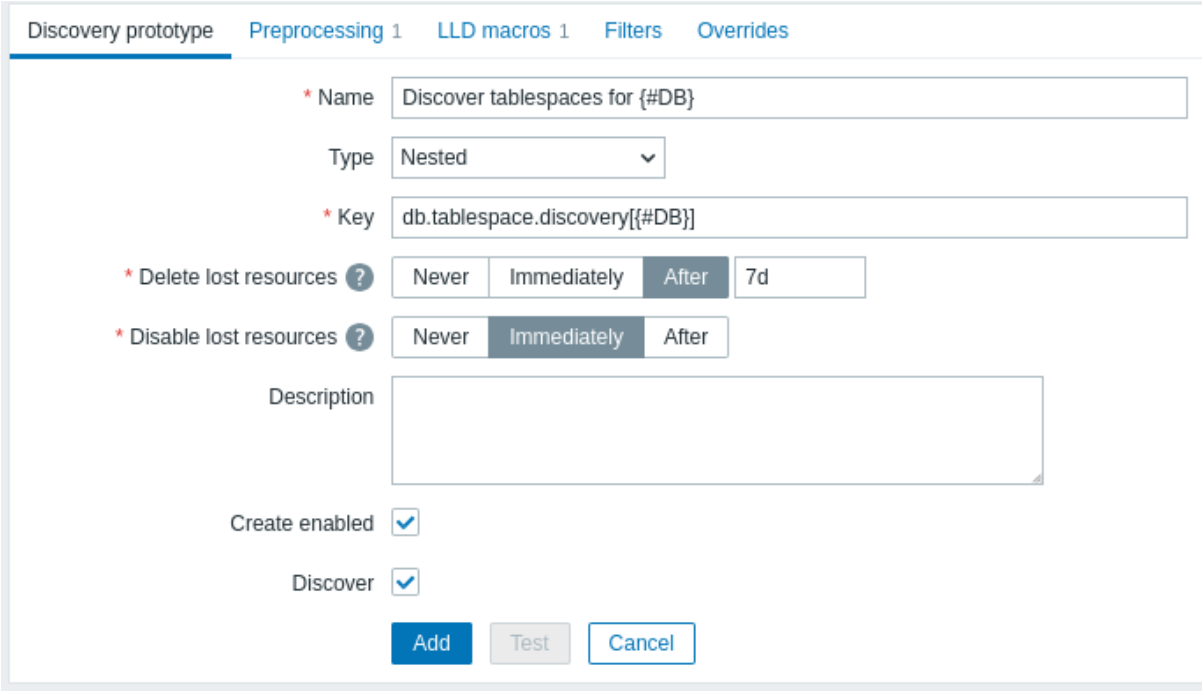
Konfiguracja

Aby utworzyć prototyp wykrywania:

- Kliknij *Prototypes wykrywania* w wierszu istniejącej reguły wykrywania



- Kliknij *Utwórz prototyp wykrywania*



Pola konfiguracyjne tego formularza są współdzielone ze standardowym wykrywaniem niskiego poziomu.

Jeśli w otwartym formularzu prototypu wykrywania jako *Type* wybierzesz opcję ”Nested”, wówczas reguły wykrywania (z prototypu wykrywania) są generowane na podstawie obiektu JSON z tej samej wartości JSON co nadrzędna reguła wykrywania. Na przykład, jeśli oryginalny JSON to [<object A>, <object B>] i istnieje jeden zagnieżdżony prototyp reguły wykrywania, zostaną wygenerowane dwie reguły wykrywania na podstawie danych odpowiednio obiektu A i obiektu B.

W takim przypadku prototyp wykrywania jest aktywowany w tym samym czasie co reguła nadrzędna. Zagnieżdżona reguła może więc korzystać z preprocessingu, aby pracować na innym ”wycinku” tych samych danych, już pobranych przez nadrzędną regułę.

Makra LLD z nadrzędnej reguły LLD są dostępne dla zagnieżdżonych reguł wykrywania.

Zagnieżdżone reguły LLD na wykrytych hostach

Zagnieżdżona reguła wykrywania niskiego poziomu może być używana w szablonie hosta przypisanym do prototypu hosta. Jeśli na wykrytym hoście istnieje *zagnieżdżona* reguła wykrywania, wówczas obiekt JSON użyty do wykrycia hosta jest również przekazywany do wszystkich reguł LLD typu zagnieżdżonego na tym hoście. Więcej szczegółów znajduje się w przykładzie.

Makra LLD z reguły wykrywania, która utworzyła host, są dostępne dla zagnieżdżonych reguł wykrywania.

Przykład

Zilustrujmy możliwe zastosowanie prototypów wykrywania na podstawie otrzymania następującego przykładu wielopoziomowego JSON.

```
[
  {
    "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" }
    ]
  }
]
```

Przypadek 1

Wykrywanie instancji baz danych na serwerze bazy danych, a następnie wykrywanie tablespace'ów dla każdej instancji.

1. Masz co najmniej jeden host powiązany z wykrywaniem serwera bazy danych.
2. Utwórz regułę LLD dla tego hosta o nazwie *Wykryj bazy danych i tablespace'y*.
3. Przejdź do zakładki *Makra LLD* w tej regule i dodaj makro `{#DB}=$.database`.
4. Dodaj prototyp pozycji dla tej reguły o nazwie *Aktywne połączenia do {#DB}* (Typ: Agent, klucz: `db.connections[{#DB}]`).
5. Zostaną wykryte pozycje powiązane z każdą bazą danych:

```
Active connections to db1, Active connections to db2, Active connections to db3.
```

6. Utwórz prototyp wykrywania dla tej reguły o nazwie *Wykryj tablespace'y dla {#DB}* (Typ: Zagnieżdżony, klucz: `db.tablespace.discovery[{#DB}]`).
7. Przejdź do zakładki *Przetwarzanie wstępne* tego prototypu wykrywania i dodaj krok `JSONPath=$.tablespaces`.
8. Przejdź do zakładki *Makra LLD* tego prototypu wykrywania i dodaj makro `{#TSNAME}=$.name`.
9. Utwórz prototyp pozycji dla tego prototypu wykrywania o nazwie *Rozmiar tablespace'a {#TSNAME} dla {#DB}* (Typ: Agent, klucz: `db.ts.size[{#DB}, {#TSNAME}]`).
10. Zostaną wykryte pozycje powiązane z każdym tablespace'em każdej bazy danych:

```
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.
```

z kluczami `db.ts.size[db1,ts1]`, `db.ts.size[db1,ts2]`, ... `db.ts.size[db3,ts3]`.

Przypadek 2

Odkrywanie instancji baz danych na serwerze bazy danych poprzez reprezentowanie ich jako odkrytych hostów, a następnie odkrywanie tablespace dla każdej instancji.

1. Masz co najmniej jeden host (host główny) powiązany z odkrywaniem serwera bazy danych.
2. Utwórz szablon do odkrywania tablespace dla każdej bazy danych.
3. Utwórz w tym szablonie pozycję o nazwie *Aktywne połączenia do {#DB}* (Typ: Agent, Klucz: `db.connections[{#DB}]`).
4. Utwórz regułę LLD dla tego szablonu o nazwie *Odkryj tablespace* (Typ: Zagnieżdżony).
5. Przejdź do zakładki *Preprocessing* tej reguły i dodaj krok `JSONPath=$.tablespaces`.
6. Przejdź do zakładki *LLD Macros* tej reguły, dodaj makro `{#TSNAME}=$.name`.

7. Utwórz prototyp pozycji dla tej reguły o nazwie *Rozmiar tablespace {#TSNAME} dla {#DB}* (Typ: Agent, Klucz: `db.ts.size[{#DB}, {#TSNAME}]`).
8. Wracając do hosta głównego, utwórz regułę LLD dla tego hosta o nazwie *Odkryj bazy danych i tablespace*.
9. Przejdź do zakładki *LLD Macros* tej reguły, dodaj makro `{#DB}=$.database`.
10. Dodaj prototyp hosta dla tej reguły o nazwie *Host dla bazy danych {#DB}*.
11. Przejdź do zakładki *Macros* tego prototypu hosta, dodaj makro `{$DB}={#DB}` (dla nazwy i klucza pozycji z kroku 3).
12. Podłącz szablon z kroku 2 do tego prototypu hosta.
13. Odkryte hosty zawierają odkryte pozycje powiązane z każdą bazą danych i jej tablespace:

Host	Pozycje
Host dla bazy danych db1	Aktywne połączenia do db1
	Rozmiar tablespace ts1 dla db1
	Rozmiar tablespace ts2 dla db1
	Rozmiar tablespace ts3 dla db1
Host dla bazy danych db2	Aktywne połączenia do db2
	Rozmiar tablespace ts1 dla db2
	Rozmiar tablespace ts2 dla db2
	Rozmiar tablespace ts3 dla db2
Host dla bazy danych db3	Aktywne połączenia do db3
	Rozmiar tablespace ts1 dla db3
	Rozmiar tablespace ts2 dla db3
	Rozmiar tablespace ts3 dla db3

Uwagi dotyczące wykrywania niskiego poziomu

Używanie makr LLD w kontekstach makr użytkownika

Makra LLD mogą być używane wewnątrz kontekstu makr użytkownika, na przykład w prototypach wyzwalaczy.

Wiele reguł LLD dla tej samej pozycji

Możliwe jest zdefiniowanie kilku reguł wykrywania niskopoziomowego z tą samą pozycją wykrywania.

W tym celu należy zdefiniować parametr Alias agent parameter, który umożliwia używanie zmodyfikowanych kluczy pozycji wykrywania w różnych regułach wykrywania, na przykład `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]` itd.

Limity danych dla wartości zwracanych

Nie ma limitu dla danych JSON reguły wykrywania niskiego poziomu, jeśli są one odbierane bezpośrednio przez serwer Zabbix. Wynika to z tego, że wartości zwracane są przetwarzane bez zapisywania ich w bazie danych.

Nie ma również limitu dla niestandardowych reguł wykrywania niskiego poziomu. Jeśli jednak dane niestandardowej reguły wykrywania niskiego poziomu są pobierane przy użyciu parametru użytkownika, obowiązuje limit wartości zwracanej dla parametru użytkownika.

Jeśli dane muszą przejść przez proxy Zabbix, proxy musi zapisać te dane w bazie danych. W takim przypadku obowiązują limity bazy danych.

Przykłady reguł wykrywania

Ta sekcja zawiera praktyczne przykłady konfiguracji reguł wykrywania dla różnych typów obiektów, takich jak systemy plików, interfejsy sieciowe, usługi, czujniki, obiekty SNMP, aplikacje Java i bazy danych.

Każdy przykład pokazuje, jak może być zbudowana reguła wykrywania, w tym użyty klucz pozycji lub zapytanie, wynik JSON z makrami LLD oraz sposób wykorzystania tych wyników do tworzenia prototypów pozycji, wyzwalaczy i wykresów.

- Wykrywanie urządzeń blokowych
- Wykrywanie procesorów i rdzeni CPU
- Wykrywanie interfejsów hosta w Zabbix
- Wykrywanie czujników IPMI
- Wykrywanie obiektów JMX
- Wykrywanie zamontowanych systemów plików

- Wykrywanie interfejsów sieciowych
- Wykrywanie identyfikatorów SNMP OID
- Wykrywanie identyfikatorów SNMP OID (starsza wersja)
- Wykrywanie usług systemd
- Wykrywanie wystąpień liczników wydajności systemu Windows
- Wykrywanie usług systemu Windows
- Wykrywanie za pomocą zapytań WMI
- Wykrywanie za pomocą zapytań SQL ODBC
- Wykrywanie z użyciem danych Prometheus

Niestandardowe reguły LLD

Przegląd

Możliwe jest również utworzenie całkowicie niestandardowej reguły LLD, wykrywającej dowolny typ encji — na przykład bazy danych na serwerze bazodanowym.

Aby to zrobić, należy utworzyć niestandardową pozycja, która zwraca ciąg JSON, określający znalezione obiekty oraz opcjonalnie — niektóre ich właściwości. Liczba makr na encję nie jest ograniczona — podczas gdy wbudowane reguły wykrywania zwracają jedno lub dwa makra (na przykład dwa w przypadku wykrywania systemów plików), możliwe jest zwrócenie większej liczby.

Przykład

Wymagany format ciągu JSON najlepiej zilustrować na przykładzie. Załóżmy, że używasz starego agenta Zabbix 1.8 (takiego, który nie obsługuje klucza `vfs.fs.discovery`), ale nadal musisz wykrywać systemy plików. Poniżej znajduje się prosty skrypt Perl dla systemu Linux, który wykrywa zamontowane systemy plików i zwraca dane w formacie JSON, zawierające zarówno nazwę systemu plików, jak i jego typ. Jednym ze sposobów użycia go byłoby zastosowanie go jako UserParameter z kluczem `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:
Dozwolone symbole w nazwach makr LLD to **0-9** , **A-Z** , **_** , **.**. Małe litery nie są obsługiwane w nazwach.

Przykład danych wyjściowych (przeformatowany dla większej czytelności) pokazano poniżej. JSON dla niestandardowych kontroli wykrywania musi mieć ten sam format.

```
[
    { "{#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"  }
]
```

W poprzednim przykładzie wymagane jest, aby klucze odpowiadały nazwom makr LLD używanym w prototypach; alternatywnie można wyodrębnić wartości makr LLD za pomocą JSONPath {#FSNAME} → $.fsname oraz {#FSTYPE} → $.fstype, co umożliwia użycie takiego skryptu:

```
####!/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";
```

Przykład danych wyjściowych (przeformatowany dla większej czytelności) pokazano poniżej.

```
[
    { "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"  }
]
```

Następnie w polu *Filter* reguły wykrywania możesz określić {#FSTYPE} jako makro, a `rootfs|ext3` jako wyrażenie regularne.

Note:
Nie musisz używać nazw makr takich jak FSNAME/FSTYPE w niestandardowych regułach LLD; możesz używać dowolnych nazw. Jeśli używany jest JSONPath, wiersz LLD będzie elementem tablicy, który może być obiektem, ale może też być inną tablicą lub wartością.

Pamiętaj, że w przypadku użycia parametru użytkownika wartość zwracana jest ograniczona do 16 MB. Więcej informacji znajdziesz w sekcji limity danych dla wartości zwracanych LLD.

Monitorowanie maszyn wirtualnych

Przegląd Zabbix może używać reguł odkrywania niskiego poziomu do automatycznego wykrywania hiperwizorów VMware i maszyn wirtualnych oraz tworzenia hostów do ich monitorowania na podstawie wstępnie zdefiniowanych prototypów hostów.

Zabbix zawiera również gotowe do użycia szablony do monitorowania VMware vCenter lub hiperwizorów ESXi.

Minimalna wymagana wersja VMware vCenter lub vSphere to 5.1.

Zbieranie danych Monitorowanie maszyn wirtualnych składa się z dwóch etapów:

1. Procesy Zabbix *vmware collector* przetwarzają dane maszyn wirtualnych — procesy te pobierają niezbędne informacje z usług sieciowych VMware za pośrednictwem protokołu SOAP, wstępnie je przetwarzają i zapisują we współdzielonej pamięci serwera Zabbix.
2. Procesy Zabbix *poller* pobierają dane przy użyciu prostego sprawdzenia Zabbix klucze pozycji monitorowania VMware.

Zabbix dzieli zebrane dane na dane konfiguracyjne VMware oraz dane liczników wydajności VMware. Oba typy danych są zbierane niezależnie przez procesy *vmware collector*.

Na podstawie informacji z liczników wydajności VMware dostępne są następujące statystyki:

- Magazyn danych
- Urządzenie dyskowe
- CPU
- Zasilanie
- Interfejs sieciowy
- Niestandardowe pozycje liczników wydajności

Pełną listę pozycji pobierających dane z liczników wydajności VMware można znaleźć w *kluczach pozycji monitorowania VMware*.

Należy pamiętać, że częstotliwość pobierania zdarzeń VMware zależy od interwału odpytywania vmware.eventlog, ale nie może być mniejsza niż 5 sekund.

Konfiguracja Jeśli serwer Zabbix jest skompilowany ze źródeł, musi zostać skompilowany z opcjami konfiguracji `--with-libcurl --with-libxml2`, aby włączyć monitorowanie maszyn wirtualnych. Pakiety Zabbix są już skompilowane z włączonymi tymi opcjami.

Następujące parametry pliku konfiguracyjnego serwera Zabbix można zmodyfikować na potrzeby monitorowania maszyn wirtualnych:

- `StartVMwareCollectors`

Note:
Zaleca się włączenie większej liczby kolektorów niż liczba monitorowanych usług VMware; w przeciwnym razie pobieranie statystyk liczników wydajności VMware może zostać opóźnione przez pobieranie danych konfiguracyjnych VMware (co w przypadku dużych instalacji trwa pewien czas).

 Ogólnie wartość `StartVMwareCollectors` nie powinna być mniejsza niż 2 ani większa niż dwukrotność liczby monitorowanych usług VMware: Liczba usług < `StartVMwareCollectors` < (Liczba usług * 2). Na przykład podczas monitorowania jednej usługi VMware ustaw `StartVMwareCollectors` na 2; podczas monitorowania trzech usług ustaw `StartVMwareCollectors` na 5.

 Należy pamiętać, że wymagana liczba kolektorów zależy również od zakresu środowiska VMware oraz od parametrów konfiguracyjnych `VMwareFrequency` i `VMwarePerfFrequency`.

- `VMwareCacheSize`
- `VMwareFrequency`
- `VMwarePerfFrequency`
- `VMwareTimeout`

Attention:
Aby obsługiwać metryki pojemności datastore, upewnij się, że wartość klucza VMware `vpxd.stats.maxQueryMetrics` jest ustawiona na co najmniej 64. Więcej informacji można znaleźć w artykule *VMware Knowledge Base*.

Odkrywanie

Zabbix może używać reguł odkrywania niskiego poziomu (na przykład `vmware.hv.discovery[{$VMWARE.URL}]`), aby automatycznie wykrywać hipernadzorców VMware i maszyny wirtualne. Ponadto Zabbix może używać prototypów hostów do automatycznego generowania rzeczywistych hostów dla wykrytych encji. Więcej informacji można znaleźć w *Prototypy hostów*.

Przykłady konfiguracji

Podstawowy przykład konfiguracji Zabbix do monitorowania VMware z użyciem szablonu *VMware FQDN* znajduje się w sekcji *Monitorowanie VMware za pomocą Zabbix*.

Bardziej szczegółowy przykład tworzenia hosta, reguły niskopoziomowego wykrywania oraz prototypu hosta do monitorowania VMware znajduje się w sekcji *Przykład konfiguracji*.

Rozszerzone rejestrowanie Dane zbierane przez procesy *vmware collector* można rejestrować na potrzeby szczegółowego debugowania przy użyciu poziomu debugowania 5. Poziom debugowania można skonfigurować w plikach konfiguracyjnych serwera i proxy lub za pomocą opcji sterowania w czasie działania `-R log_level_increase="vmware collector,N"`, gdzie `N` oznacza numer procesu.

Na przykład aby zwiększyć poziom debugowania z 4 do 5 dla wszystkich procesów *vmware collector*, uruchom następujące polecenie:

```
zabbix_server -R log_level_increase="vmware collector"
```

Aby zwiększyć poziom debugowania z 4 do 5 dla drugiego procesu *vmware collector*, uruchom następujące polecenie:

```
zabbix_server -R log_level_increase="vmware collector,2"
```

Gdy rozszerzone rejestrowanie danych VMware nie jest już wymagane, zaleca się zmniejszenie poziomu debugowania do wartości domyślnej (3) za pomocą polecenia `-R log_level_decrease`.

Rozwiązywanie problemów

- W przypadku niedostępnych metryk upewnij się, że nie zostały one w nowszych wersjach VMware vSphere wyłączone lub ustawione jako niedostępne domyślnie, albo że nie nałożono ograniczeń na zapytania do bazy danych metryk wydajności. Więcej informacji można znaleźć w ZBX-12094.
- Jeśli `config.vpxd.stats.maxQueryMetrics` jest nieprawidłowe lub przekracza maksymalną dozwoloną liczbę znaków, dodaj parametr `config.vpxd.stats.maxQueryMetrics` do ustawień vCenter Server. Wartość tego parametru powinna być taka sama jak wartość `maxQuerysize` w pliku `web.xml` firmy VMware. Więcej informacji można znaleźć w artykule *VMware Knowledge Base*.
- Jeśli podejrzewasz, że Twoja instalacja Zabbix zużywa zbyt dużo pamięci, zobacz Profiling excessive memory usage with tcmalloc.

Klucze pozycji monitorowania VMware

Przegląd Ta strona zawiera szczegółowe informacje o prostych kontrolach, które mogą być używane do monitorowania środowisk VMware. Metryki są pogrupowane według celu monitorowania.

Obsługiwane klucze pozycji Klucze pozycji są wymienione bez parametrów i dodatkowych informacji. Kliknij klucz pozycji, aby zobaczyć pełne szczegóły.

Item key	Description	Item group
vmware.eventlog	Dziennik zdarzeń VMware.	General service
vmware.fullname	Pełna nazwa usługi VMware.	
vmware.version	Wersja usługi VMware.	
vmware.cl.perfcounter	Metryki licznika wydajności klastra VMware.	Cluster
vmware.cluster.alarms.get	Dane alarmów klastra VMware.	
vmware.cluster.discovery	Wykrywanie klastrów VMware.	
vmware.cluster.property	Właściwość klastra VMware.	
vmware.cluster.status	Stan klastra VMware.	
vmware.cluster.tags.get	Tablica tagów klastra VMware.	
vmware.datastore.alarms.get	Dane alarmów magazynu danych VMware.	Datastore
vmware.datastore.discovery	Wykrywanie magazynów danych VMware.	
vmware.datastore.hv.list	Lista hypervisorów magazynu danych.	
vmware.datastore.perfcounter	Wartość licznika wydajności magazynu danych VMware.	
vmware.datastore.property	Właściwość magazynu danych VMware.	
vmware.datastore.read	Czas operacji odczytu z magazynu danych.	
vmware.datastore.size	Miejsce w magazynie danych VMware w bajtach lub jako procent całości.	
vmware.datastore.tags.get	Tablica tagów magazynu danych VMware.	
vmware.datastore.write	Czas operacji zapisu do magazynu danych.	

Item key	Description	Item group
vmware.dc.alarms.get	Dane alarmów centrum danych VMware.	Datacenter
vmware.dc.discovery	Wykrywanie centrów danych VMware.	
vmware.dc.tags.get	Tablica tagów centrum danych VMware.	
vmware.dvswitch.discovery	Wykrywanie rozproszonych przełączników VMware vSphere.	vSphere Distributed Switch
vmware.dvswitch.fetchportsget	Dane portów rozproszonego przełącznika VMware vSphere.	
vmware.hv.alarms.get	Dane alarmów hypervisora VMware.	Hypervisor
vmware.hv.cluster.name	Nazwa klastra hypervisora VMware.	
vmware.hv.connectionstate	Stan połączenia hypervisora VMware.	
vmware.hv.cpu.usage	Użycie procesora hypervisora VMware (Hz).	
vmware.hv.cpu.usage.perf	Użycie procesora hypervisora VMware jako procent w danym interwale.	
vmware.hv.cpu.utilization	Użycie procesora hypervisora VMware jako procent w danym interwale, zależne od zarządzania energią lub HT.	
vmware.hv.datacenter.name	Nazwa centrum danych hypervisora VMware.	
vmware.hv.datastore.discovery	Wykrywanie magazynów danych hypervisora VMware.	
vmware.hv.datastore.list	Lista magazynów danych hypervisora VMware.	
vmware.hv.datastore.multipath	Liczba dostępnych ścieżek magazynu danych.	
vmware.hv.datastore.read	Średni czas operacji odczytu z magazynu danych.	
vmware.hv.datastore.size	Miejsce w magazynie danych VMware w bajtach lub jako procent całości.	
vmware.hv.datastore.write	Średni czas operacji zapisu do magazynu danych.	
vmware.hv.discovery	Wykrywanie hypervisorów VMware.	
vmware.hv.diskinfo.get	Dane dysków hypervisora VMware.	
vmware.hv.fullname	Pełna nazwa produktu, w tym informacje o wersji.	
vmware.hv.hw.cpu.freq	Częstotliwość procesora hypervisora VMware.	
vmware.hv.hw.cpu.model	Model procesora hypervisora VMware.	
vmware.hv.hw.cpu.num	Liczba rdzeni procesora na hypervisorze VMware.	
vmware.hv.hw.cpu.threads	Liczba wątków procesora na hypervisorze VMware.	
vmware.hv.hw.memory	Całkowity rozmiar pamięci hypervisora VMware.	
vmware.hv.hw.model	Model hypervisora VMware.	
vmware.hv.hw.sensors.get	Wartość czujników sprzętowych hypervisora VMware.	
vmware.hv.hw.serialnumber	Numer seryjny hypervisora VMware.	
vmware.hv.hw.uuid	UUID BIOS hypervisora VMware.	
vmware.hv.hw.vendor	Nazwa producenta hypervisora VMware.	
vmware.hv.maintenance	Stan konserwacji hypervisora VMware.	
vmware.hv.memory.size.ballooned	Rozmiar pamięci ballooned hypervisora VMware.	
vmware.hv.memory.used	Rozmiar użytej pamięci hypervisora VMware.	
vmware.hv.net.if.discovery	Wykrywanie interfejsów sieciowych hypervisora VMware.	
vmware.hv.network.in	Statystyki wejściowe sieci hypervisora VMware.	
vmware.hv.network.linkspeed	Prędkość interfejsu sieciowego hypervisora VMware.	
vmware.hv.network.out	Statystyki wyjściowe sieci hypervisora VMware.	
vmware.hv.perfcounter	Wartość licznika wydajności hypervisora VMware.	
vmware.hv.property	Właściwość hypervisora VMware.	
vmware.hv.power	Zużycie energii przez hypervisor VMware.	
vmware.hv.sensor.health.state	Stan zbiorczy czujnika kondycji hypervisora VMware.	
vmware.hv.sensors.get	Czujniki stanu dostawcy HW hypervisora VMware.	
vmware.hv.status	Stan hypervisora VMware.	
vmware.hv.tags.get	Tablica tagów hypervisora VMware.	
vmware.hv.uptime	Czas działania hypervisora VMware.	
vmware.hv.version	Wersja hypervisora VMware.	
vmware.hv.vm.num	Liczba maszyn wirtualnych na hypervisorze VMware.	
vmware.rp.cpu.usage	Użycie CPU w hercach w danym interwale w VMware Resource Pool.	Resource pool
vmware.rp.memory	Metryki pamięci zasobu VMware resource pool.	
vmware.alarms.get	Dane alarmów VMware virtual center.	Virtual center
vmware.vm.alarms.get	Dane alarmów maszyny wirtualnej VMware.	Virtual machine
vmware.vm.attribute	Wartość niestandardowego atrybutu maszyny wirtualnej VMware.	
vmware.vm.cluster.name	Nazwa maszyny wirtualnej VMware.	
vmware.vm.consolidationneeded	Czy maszyny wirtualnej VMware wymaga konsolidacji.	

Item key	Description	Item group
vmware.vm.cpu.latency	Procent czasu, w którym maszyna wirtualna nie może działać, ponieważ konkuruje o dostęp do fizycznego procesora(ów).	
vmware.vm.cpu.num	Liczba procesorów na maszynie wirtualnej VMware.	
vmware.vm.cpu.readiness	Procent czasu, w którym maszyna wirtualna była gotowa, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym procesorze.	
vmware.vm.cpu.ready	Czas, w którym maszyna wirtualna była gotowa, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym procesorze.	
vmware.vm.cpu.swapwait	Procent czasu CPU spędzony na oczekiwaniu na swap-in.	
vmware.vm.cpu.usage	Użycie procesora maszyny wirtualnej VMware (Hz).	
vmware.vm.cpu.usage.perf	Użycie procesora maszyny wirtualnej VMware jako procent w danym interwale.	
vmware.vm.datacenter.name	Nazwa centrum danych maszyny wirtualnej VMware.	
vmware.vm.discovery	Wykrywanie maszyn wirtualnych VMware.	
vmware.vm.guest.memory.swapped	Ilość pamięci fizycznej gościa, która została przeniesiona do przestrzeni wymiany.	
vmware.vm.guest.osuptime	Łączny czas, jaki upłynął od ostatniego uruchomienia systemu operacyjnego.	
vmware.vm.hv.maintenance	Stan konserwacji hypervisora maszyny wirtualnej VMware.	
vmware.vm.hv.name	Nazwa hypervisora maszyny wirtualnej VMware.	
vmware.vm.memory.size	Całkowity rozmiar pamięci maszyny wirtualnej VMware.	
vmware.vm.memory.size.ballooned	Rozmiar pamięci ballooned maszyny wirtualnej VMware.	
vmware.vm.memory.size.compressed	Rozmiar skompresowanej pamięci maszyny wirtualnej VMware.	
vmware.vm.memory.size.consumed	Ilość fizycznej pamięci hosta zużytej do buforowania stron fizycznej pamięci gościa.	
vmware.vm.memory.size.private	Rozmiar prywatnej pamięci maszyny wirtualnej VMware.	
vmware.vm.memory.size.shared	Rozmiar współdzielonej pamięci maszyny wirtualnej VMware.	
vmware.vm.memory.size.swapped	Rozmiar pamięci wymienionej maszyny wirtualnej VMware.	
vmware.vm.memory.size.usage.guest	Użycie pamięci gościa maszyny wirtualnej VMware.	
vmware.vm.memory.size.usage.host	Użycie pamięci hosta maszyny wirtualnej VMware.	
vmware.vm.memory.usage	Procent zużytej fizycznej pamięci hosta.	
vmware.vm.net.if.discovery	Wykrywanie interfejsów sieciowych maszyny wirtualnej VMware.	
vmware.vm.net.if.in	Statystyki wejściowe interfejsu sieciowego maszyny wirtualnej VMware.	
vmware.vm.net.if.out	Statystyki wyjściowe interfejsu sieciowego maszyny wirtualnej VMware.	
vmware.vm.net.if.usage	Wykorzystanie sieci maszyny wirtualnej VMware w danym interwale.	
vmware.vm.perfcounter	Wartość licznika wydajności maszyny wirtualnej VMware.	
vmware.vm.powerstate	Stan zasilania maszyny wirtualnej VMware.	
vmware.vm.property	Właściwość maszyny wirtualnej VMware.	
vmware.vm.snapshot.get	Stan migawki maszyny wirtualnej VMware.	
vmware.vm.state	Stan maszyny wirtualnej VMware.	
vmware.vm.storage.committed	Zadeklarowana przestrzeń dyskowa maszyny wirtualnej VMware.	
vmware.vm.storage.readoio	Średnia liczba oczekujących żądań odczytu do dysku wirtualnego w czasie interwału zbierania.	
vmware.vm.storage.totalreadlatency	Średni czas odczytu z dysku wirtualnego.	
vmware.vm.storage.totalwritelatency	Średni czas zapisu na dysk wirtualny.	
vmware.vm.storage.uncommitted	Niezadeklarowana przestrzeń dyskowa maszyny wirtualnej VMware.	
vmware.vm.storage.unshared	Niewspółdzielona przestrzeń dyskowa maszyny wirtualnej VMware.	
vmware.vm.storage.writeoio	Średnia liczba oczekujących żądań zapisu do dysku wirtualnego w czasie interwału zbierania.	
vmware.vm.tags.get	Tablica tagów maszyny wirtualnej VMware.	
vmware.vm.tools	Stan lub wersja narzędzi gościa maszyny wirtualnej VMware.	
vmware.vm.uptime	Czas działania maszyny wirtualnej VMware.	
vmware.vm.vfs.dev.discovery	Wykrywanie urządzeń dyskowych maszyny wirtualnej VMware.	
vmware.vm.vfs.dev.read	Statystyki odczytu urządzenia dyskowego maszyny wirtualnej VMware.	
vmware.vm.vfs.dev.write	Statystyki zapisu urządzenia dyskowego maszyny wirtualnej VMware.	
vmware.vm.vfs.fs.discovery	Wykrywanie systemów plików maszyny wirtualnej VMware.	
vmware.vm.vfs.fs.size	Statystyki systemu plików maszyny wirtualnej VMware.	

Szczegóły klucza pozycji Parametry bez nawiasów ostrych są obowiązkowe. Parametry oznaczone nawiasami ostrymi **< >** są opcjonalne.

vmware.eventlog[url,<mode>,<severity>]

 Dziennik zdarzeń VMware.
 Zwracana wartość: *Log*.

Parametry:

- **url** - URL usługi VMware;

- **mode** - *all* (domyślnie) lub *skip* - pomiń przetwarzanie starszych danych;
- **severity** - filtrowanie według poziomu ważności: *error*, *warning*, *info* lub *user*. Ten parametr musi być ujęty w cudzysłów, jeśli określono więcej niż jeden poziom ważności na liście rozdzielanej przecinkami (np.`"error,warning,info,user"`). Domyślnie wyłączone.

Uwagi:

- Dla każdego URL może istnieć tylko jeden klucz pozycji `vmware.eventlog`;
- Zobacz także przykład filtrowania rekordów dziennika zdarzeń VMware.

vmware.fullnameurl

 Pełna nazwa usługi VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware.

vmware.versionurl

 Wersja usługi VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - URL usługi VMware.

vmware.cl.perfcounter[url,id,path,<instance>]

 Metryki licznika wydajności klastra VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - URL usługi VMware;
- **id** - ID klastra VMware. `id` można uzyskać z `vmware.cluster.discovery[]` jako {#CLUSTER.ID}.
- **path** - ścieżka licznika wydajności[1];
- **instance** - instancja licznika wydajności.

vmware.cluster.alarms.get[url,id]

 Dane alarmów klastra VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator klastra VMware.

vmware.cluster.discoveryurl

 Wykrywanie klastrów VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.cluster.property[url,id,prop]

 Właściwość klastra VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **id** — identyfikator klastra VMware;
- **prop** — ścieżka właściwości, będąca nazwą właściwości obiektu VM zgodnie z definicją w VMware SDK.

Przykłady:

```
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},overallStatus]
vmware.vm.property[{$VMWARE.URL},{$VMWARE.VM.UUID},runtime.powerState]
```

vmware.cluster.status[url,name]

 Status klastra VMware.
 Zwracana wartość: *0* - szary; *1* - zielony; *2* - żółty; *3* - czerwony.

Parametry:

- **url** - adres URL usługi VMware;
- **name** - nazwa klastra VMware.

vmware.cluster.tags.get[url,id]

 Tablica tagów klastra VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator klastra VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.datastore.alarms.get[url,uuid]

 Dane alarmów magazynu danych VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator magazynu danych VMware.

vmware.datastore.discovery[url,<filter_uuid>]

 Odkrywanie magazynów danych VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **filter_uuid** — pojedynczy unikalny identyfikator HV lub VM (domyślnie: pusty). Jeśli zostanie określony, zostaną odkryte tylko powiązane magazyny danych.

vmware.datastore.hv.list[url,datastore]

 Lista hypervisorów datastore.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **datastore** - UUID lub nazwa datastore.

Przykład danych wyjściowych:

```
esx7-01-host.zabbix.sandbox
esx7-02-host.zabbix.sandbox
```

vmware.datastore.perfcounter[url,uuid,path,<instance>]

 Wartość licznika wydajności magazynu danych VMware.
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalnie unikalny identyfikator magazynu danych VMware;
- **path** — ścieżka licznika wydajności[1];
- **instance** — instancja licznika wydajności. Dla wartości zagregowanych (domyślnie) użyj pustej instancji. Wartość `instance` można otrzymać z `vmware.datastore.discovery[]` jako część tablicy {#DATASTORE.EXTENT}.

vmware.datastore.property[url,uuid,prop]

 Właściwość magazynu danych VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator magazynu danych VMware;
- **prop** — ścieżka właściwości.

vmware.datastore.read[url,datastore,<mode>]

 Czas trwania operacji odczytu z datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **datastore** — UUID lub nazwa datastore;
- **mode** — *latency* (wartość średnia, domyślnie) lub *maxlatency* (wartość maksymalna).

vmware.datastore.size[url,datastore,<mode>]

 Przestrzeń VMware datastore w bajtach lub jako wartość procentowa całkowitej przestrzeni.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.

Parametry:

- **url** - adres URL usługi VMware;
- **datastore** - UUID lub nazwa datastore;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *pfree* (procent wolnej przestrzeni), *uncommitted*.

vmware.datastore.tags.get[url,uuid]

 Tablica tagów datastore VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator datastore VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.datastore.write[url,datastore,<mode>]

 Czas trwania operacji zapisu do datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - URL usługi VMware;
- **datastore** - UUID lub nazwa datastore;
- **mode** - *latency* (wartość średnia, domyślnie) lub *maxlatency* (wartość maksymalna).

vmware.dc.alarms.get[url,id]

 Dane alarmów centrum danych VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator centrum danych VMware.

vmware.dc.discoveryurl

 Wykrywanie centrów danych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware.

vmware.dc.tags.get[url,id]

 Tablica tagów centrum danych VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **id** - identyfikator centrum danych VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.dvswitch.discoveryurl

 Wykrywanie przełączników rozproszonych VMware vSphere.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.dvswitch.fetchports.get[url,uuid,<filter>,<mode>]

 Dane portów VMware vSphere Distributed Switch.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator VMware DVSwitch.
- **filter** - pojedynczy ciąg znaków z kryteriami rozdzielonymi przecinkami do wybierania portów;
- **mode** - *state* (wszystkie XML bez węzłów XML ”config”, domyślnie) lub *full*.

Parametr **filter** obsługuje kryteria dostępne w obiekcie danych VMware DistributedVirtualSwitchPortCriteria.

Przykład:

```
vmware.dvswitch.fetchports.get[{$VMWARE.URL},{$VMWARE.DVS.UUID},"connected:true,active:true,uplinkPort:fal
```

vmware.hv.alarms.get[url,uuid]

 Dane alarmów hipernadzorcy VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.cluster.name[url,uuid]

 Nazwa klastra hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.connectionstate[url,uuid]

 Stan połączenia z hiperwizorem VMware.
 Zwracana wartość: *String*: *connected*, *disconnected* lub *notResponding*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.cpu.usage[url,uuid]

 Użycie procesora hypervisora VMware (Hz).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.cpu.usage.perf[url,uuid]

 Użycie procesora hypervisora VMware jako wartość procentowa w danym przedziale czasu.
 Zwracana wartość: *Float*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.cpu.utilization[url,uuid]

 Użycie procesora hipernadzorcy VMware jako wartość procentowa w danym przedziale czasu; zależy od zarządzania energią lub HT.
 Zwracana wartość: *Float*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.datacenter.name[url,uuid]

 Nazwa centrum danych hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.datastore.discovery[url,uuid]

 Wykrywanie magazynów danych hipernadzorcy VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.datastore.list[url,uuid]

 Lista magazynów danych hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

Przykład danych wyjściowych:

```
SSD-RAID1-VAULT1
SSD-RAID1-VAULT2
SSD-RAID10
```

vmware.hv.datastore.multipath[url,uuid,<datastore>,<partitionid>]

 Liczba dostępnych ścieżek datastore.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware;
- **datastore** - UUID lub nazwa datastore;
- **partitionid** - wewnętrzny identyfikator urządzenia fizycznego z `vmware.hv.datastore.discovery`.

vmware.hv.datastore.read[url,uuid,datastore,<mode>]

 Średni czas trwania operacji odczytu z datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;
- **datastore** — UUID lub nazwa datastore;
- **mode** — *latency* (domyślnie).

vmware.hv.datastore.size[url,uuid,datastore,<mode>]

 Miejsce w VMware datastore w bajtach lub jako wartość procentowa całkowitej pojemności.
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowej.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware;
- **datastore** - UUID lub nazwa datastore;
- **mode** - możliwe wartości: *total* (domyślnie), *free*, *pfree* (procent wolnego miejsca), *uncommitted*.

vmware.hv.datastore.write[url,uuid,datastore,<mode>]

 Średni czas trwania operacji zapisu do datastore (w milisekundach).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;
- **datastore** — UUID lub nazwa datastore;
- **mode** — *latency* (domyślnie).

vmware.hv.discoveryurl

 Wykrywanie hiperwizorów VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware.

vmware.hv.diskinfo.get[url,uuid]

 Dane dysku hipernadzorcy VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.fullname[url,uuid]

 Pełna nazwa produktu, włącznie z informacją o wersji.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.hw.cpu.freq[url,uuid]

 Częstotliwość procesora hiperwizora VMware (Hz).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.cpu.model[url,uuid]

 Model procesora hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.cpu.num[url,uuid]

 Liczba rdzeni procesora w hiperwizorze VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.cpu.threads[url,uuid]

 Liczba wątków procesora w hiperwizorze VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.hw.memory[url,uuid]

 Całkowity rozmiar pamięci hipernadzorcy VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.hw.model[url,uuid]

 Model hypervisora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hypervisora VMware.

vmware.hv.hw.sensors.get[url,uuid]

 Wartość czujników sprzętowych hipernadzorcy VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.hw.serialnumber[url,uuid]

 Numer seryjny hipernadzorcy VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja działa z API vSphere 6.7 i nowszym.

vmware.hv.hw.uuid[url,uuid]

 UUID BIOS hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

Ta pozycja działa z API vSphere 6.7 i nowszym.

vmware.hv.hw.vendor[url,uuid]

 Nazwa dostawcy hypervisora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hypervisora VMware.

Ta pozycja działa z API vSphere 6.7 i nowszym.

vmware.hv.maintenance[url,uuid]

 Status trybu konserwacji hipernadzorcy VMware.
 Zwracana wartość: *0* - nie jest w trybie konserwacji; *1* - jest w trybie konserwacji.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.memory.size.ballooned[url,uuid]

 Rozmiar pamięci ballooned hipernadzorcy VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.memory.used[url,uuid]

 Rozmiar używanej pamięci hipernadzorcy VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.net.if.discovery[url,uuid]

 Wykrywanie interfejsów sieciowych hypervisora VMware.
 Zwracana wartość: *obiekt JSON*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

vmware.hv.network.in[url,uuid,<mode>]

 Statystyki ruchu przychodzącego sieci hipernadzorcy VMware (bajty na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;
- **mode** — *bps* (domyślnie), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.network.linkspeed[url,uuid,ifname]

 Prędkość interfejsu sieciowego hipernadzorcy VMware.
 Wartość zwracana: *Integer*. Zwraca *0*, jeśli interfejs sieciowy jest wyłączony, w przeciwnym razie zwraca wartość prędkości interfejsu.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware;

- **ifname** — nazwa interfejsu.

vmware.hv.network.out[url,uuid,<mode>]

 Statystyki wyjścia sieciowego hypervisora VMware (bajty na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware;
- **mode** - *bps* (domyślnie), *packets*, *dropped*, *errors*, *broadcast*.

vmware.hv.perfcounter[url,uuid,path,<instance>]

 Wartość licznika wydajności hipernadzorcy VMware.
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware;
- **path** - ścieżka licznika wydajności [1];
- **instance** - instancja licznika wydajności. Użyj pustej instancji dla wartości zagregowanych (domyślnie).

vmware.hv.property[url,uuid,prop]

 Właściwość hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware;
- **prop** — ścieżka właściwości.

vmware.hv.power[url,uuid,<max>]

 Zużycie energii hiperwizora VMware (W).
 Wartość zwracana: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware;
- **max** - maksymalne dozwolone zużycie energii.

vmware.hv.sensor.health.state[url,uuid]

 Sensor zbiorczego stanu kondycji hiperwizora VMware.
 Zwracana wartość: *Integer*: *0* - szary; *1* - zielony; *2* - żółty; *3* - czerwony.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware.

Należy pamiętać, że ta pozycja może nie działać w VMware vSphere 6.5 i nowszych wersjach, ponieważ VMware wycofał sensor *VMware Rollup Health State*.

vmware.hv.sensors.get[url,uuid]

 Czujniki stanu dostawcy sprzętu (HW) hipernadzorcy VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

vmware.hv.status[url,uuid]

 Status hipernadzorcy VMware.
 Zwracana wartość: *Integer*: *0* - szary; *1* - zielony; *2* - żółty; *3* - czerwony.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja używa właściwości ogólnego stanu systemu hosta.

vmware.hv.tags.get[url,uuid]

 Tablica tagów hypervisora VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hypervisora VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.hv.uptime[url,uuid]

 Czas działania hipernadzorcy VMware (w sekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hipernadzorcy VMware.

Ta pozycja używa właściwości ogólnego stanu systemu hosta.

vmware.hv.version[url,uuid]

 Wersja hiperwizora VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator hiperwizora VMware.

vmware.hv.vm.num[url,uuid]

 Liczba maszyn wirtualnych na hiperwizorze VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator hiperwizora VMware.

vmware.rp.cpu.usage[url,rpid]

 Użycie CPU w hercach w danym przedziale czasu dla puli zasobów VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **rpid** — identyfikator puli zasobów VMware.

vmware.rp.memory[url,rpid,<mode>]

 Metryki pamięci puli zasobów VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **rpid** - identyfikator puli zasobów VMware;
- **mode** - możliwe wartości:
consumed (domyślnie) - ilość fizycznej pamięci hosta zużywanej do obsługi stron fizycznej pamięci gościa
 ballooned - ilość fizycznej pamięci gościa odzyskanej z maszyny wirtualnej przez sterownik balloon w systemie gościa
overhead - fizyczna pamięć hosta zużywana przez struktury danych ESXi do uruchamiania maszyn wirtualnych

vmware.alarms.geturl

 Dane alarmów VMware virtual center.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.vm.alarms.get[url,uuid]

 Dane alarmów maszyny wirtualnej VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.attribute[url,uuid,name]

 Wartość atrybutu niestandardowego maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **name** - nazwa atrybutu niestandardowego.

vmware.vm.cluster.name[url,uuid]

 Nazwa maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.consolidationneeded[url,uuid]

 Dysk maszyny wirtualnej VMware wymaga konsolidacji.
 Zwracana wartość: *String*: *true* - konsolidacja jest wymagana; *false* - konsolidacja nie jest wymagana.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.latency[url,uuid]

 Procent czasu, w którym maszyna wirtualna nie może działać, ponieważ konkuruje o dostęp do fizycznych procesorów CPU.
 Wartość zwracana: *Float*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.num[url,uuid]

 Liczba procesorów w maszynie wirtualnej VMware.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.readiness[url,uuid,<instance>]

 Procent czasu, w którym maszyna wirtualna była gotowa, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym CPU.
 Zwracana wartość: *Float*.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja CPU.

vmware.vm.cpu.ready[url,uuid]

 Czas (w milisekundach), przez który maszyna wirtualna była gotowa, ale nie mogła zostać zaplanowana do uruchomienia na fizycznym CPU. Czas gotowości CPU zależy od liczby maszyn wirtualnych na hoście i ich obciążeń CPU (%).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.swapwait[url,uuid,<instance>]

 Procent czasu CPU spędzonego na oczekiwaniu na swap-in.
 Zwracana wartość: *Float*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja CPU.

vmware.vm.cpu.usage[url,uuid]

 Użycie procesora maszyny wirtualnej VMware (Hz).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.cpu.usage.perf[url,uuid]

 Użycie procesora maszyny wirtualnej VMware jako wartość procentowa w danym przedziale czasu.
 Zwracana wartość: *Float*.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.datacenter.name[url,uuid]

 Nazwa centrum danych maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.discoveryurl

 Wykrywanie maszyn wirtualnych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware.

vmware.vm.guest.memory.size.swapped[url,uuid]

 Ilość fizycznej pamięci gościa przeniesionej do przestrzeni wymiany (swap) (KB).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.guest.osuptime[url,uuid]

 Całkowity czas, jaki upłynął od ostatniego uruchomienia systemu operacyjnego (w sekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.hv.maintenance[url,uuid]

 Status konserwacji hypervisora maszyny wirtualnej VMware.
 Zwracana wartość: *0* - nie jest w trybie konserwacji; *1* - jest w trybie konserwacji.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.hv.name[url,uuid]

 Nazwa hypervisora maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size[url,uuid]

 Całkowity rozmiar pamięci maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.ballooned[url,uuid]

 Rozmiar pamięci ballooned maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.compressed[url,uuid]

 Rozmiar skompresowanej pamięci maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.consumed[url,uuid]

 Ilość fizycznej pamięci hosta zużywanej do obsługi stron fizycznej pamięci gościa (KB).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.private[url,uuid]

 Rozmiar pamięci prywatnej maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.shared[url,uuid]

 Rozmiar pamięci współdzielonej maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Liczba całkowita*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.swapped[url,uuid]

 Rozmiar pamięci wymienionej do przestrzeni wymiany maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.usage.guest[url,uuid]

 Użycie pamięci gościa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.size.usage.host[url,uuid]

 Użycie pamięci hosta maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.memory.usage[url,uuid]

 Procent pamięci fizycznej hosta, która została zużyta.
 Zwracana wartość: *Float*.

Parametry:

- **url** - adres URL usługi VMware;

- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.net.if.discovery[url,uuid]

 Wykrywanie interfejsów sieciowych maszyn wirtualnych VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.net.if.in[url,uuid,instance,<mode>]

 Statystyki wejściowe interfejsu sieciowego maszyny wirtualnej VMware (bajty/pakiety na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja interfejsu sieciowego;
- **mode** - *bps* (domyślnie) lub *pps* - bajty lub pakiety na sekundę.

vmware.vm.net.if.out[url,uuid,instance,<mode>]

 Statystyki ruchu wychodzącego interfejsu sieciowego maszyny wirtualnej VMware (bajty/pakiety na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja interfejsu sieciowego;
- **mode** — *bps* (domyślnie) lub *pps* — bajty lub pakiety na sekundę.

vmware.vm.net.if.usage[url,uuid,<instance>]

 Wykorzystanie sieci maszyny wirtualnej VMware (łączne szybkości wysyłania i odbierania) w danym przedziale czasu (KBps).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja interfejsu sieciowego.

vmware.vm.perfcounter[url,uuid,path,<instance>]

 Wartość licznika wydajności maszyny wirtualnej VMware.
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **path** - ścieżka licznika wydajności [1];
- **instance** - instancja licznika wydajności. Dla wartości zagregowanych (domyślnie) użyj pustej instancji.

vmware.vm.powerstate[url,uuid]

 Stan zasilania maszyny wirtualnej VMware.
 Zwracana wartość: *0* - poweredOff; *1* - poweredOn; *2* - suspended.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.property[url,uuid,prop]

 Właściwość maszyny wirtualnej VMware.
 Zwracana wartość: *String*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **prop** - ścieżka właściwości.

vmware.vm.snapshot.get[url,uuid]

 Stan migawki maszyny wirtualnej VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.state[url,uuid]

 Stan maszyny wirtualnej VMware.
 Zwracana wartość: *String*: *notRunning*, *resetting*, *running*, *shuttingDown*, *standby* lub *unknown*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.committed[url,uuid]

 Zajęta przestrzeń dyskowa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.readoio[url,uuid,instance]

 Średnia liczba oczekujących żądań odczytu do dysku wirtualnego w czasie interwału zbierania danych.
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** - instancja urządzenia dyskowego.

vmware.vm.storage.totalreadlatency[url,uuid,instance]

 Średni czas trwania odczytu z dysku wirtualnego (w milisekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego.

vmware.vm.storage.totalwritelatency[url,uuid,instance]

 Średni czas trwania zapisu na dysk wirtualny (w milisekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego.

vmware.vm.storage.uncommitted[url,uuid]

 Niezatwierdzona przestrzeń dyskowa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.unshared[url,uuid]

 Niewspółdzielona przestrzeń dyskowa maszyny wirtualnej VMware (w bajtach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.storage.writeoio[url,uuid,instance]

 Średnia liczba oczekujących żądań zapisu do dysku wirtualnego w czasie interwału zbierania danych.
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego.

vmware.vm.tags.get[url,uuid]

 Tablica tagów maszyny wirtualnej VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware.

Ta pozycja działa z vSphere 6.5 i nowszymi wersjami.

vmware.vm.tools[url,uuid,<mode>]

 Stan narzędzi gościa maszyny wirtualnej VMware.
 Zwracana wartość: *String*.
 W trybie *status*: *guestToolsExecutingScripts* - VMware Tools uruchamia się; *guestToolsNotRunning* - VMware Tools nie jest uruchomione; *guestToolsRunning* - VMware Tools jest uruchomione.
 W trybie *version*: wersja.

Parametry:

- **url** - URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **mode** - *version*, *status* (domyślnie).

vmware.vm.uptime[url,uuid]

 Czas działania maszyny wirtualnej VMware (w sekundach).
 Zwracana wartość: *Integer*.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.vfs.dev.discovery[url,uuid]

 Wykrywanie urządzeń dyskowych maszyny wirtualnej VMware.
 Zwracana wartość: obiekt JSON.

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

vmware.vm.vfs.dev.read[url,uuid,instance,<mode>]

 Statystyki odczytu urządzenia dyskowego maszyny wirtualnej VMware (bajty/operacje na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego;
- **mode** — *bps* (domyślnie) lub *ops* — bajty lub operacje na sekundę.

vmware.vm.vfs.dev.write[url,uuid,instance,<mode>]

 Statystyki zapisu urządzenia dyskowego maszyny wirtualnej VMware (bajty/operacje na sekundę).
 Zwracana wartość: *Integer* [2].

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware;
- **instance** — instancja urządzenia dyskowego;
- **mode** — *bps* (domyślnie) lub *ops* — bajty lub operacje na sekundę.

vmware.vm.vfs.fs.discovery[url,uuid]

 Wykrywanie systemów plików maszyny wirtualnej VMware.
 Wartość zwracana: obiekt JSON.

Parametry:

- **url** — adres URL usługi VMware;
- **uuid** — globalny unikalny identyfikator maszyny wirtualnej VMware.

Aby ta pozycja działała, w gościnnej maszynie wirtualnej muszą być zainstalowane VMware Tools.

vmware.vm.vfs.fs.size[url,uuid,fsname,<mode>]

 Statystyki systemu plików maszyny wirtualnej VMware (bajty/procenty).
 Zwracana wartość: *Integer* - dla bajtów; *Float* - dla wartości procentowych.

Parametry:

- **url** - adres URL usługi VMware;
- **uuid** - globalny unikalny identyfikator maszyny wirtualnej VMware;
- **fsname** - nazwa systemu plików;
- **mode** - *total*, *free*, *used*, *pfree* lub *pused*.

Aby ta pozycja działała, na gościnnej maszynie wirtualnej muszą być zainstalowane VMware Tools.

Przypisy

[1] Zobacz Tworzenie niestandardowych nazw liczników wydajności dla VMware.

[2] Wartość tych pozycji jest pobierana z liczników wydajności VMware, a parametr VMwarePerfFrequency parametr jest używany do odświeżania ich danych w pamięci podręcznej VMware 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

Więcej informacji

Szczegółowe informacje o tym, jak skonfigurować Zabbix do monitorowania środowisk VMware, można znaleźć w sekcji Monitorowanie maszyn wirtualnych.

Klucze wykrywania maszyn wirtualnych

Poniższa tabela zawiera pola zwracane przez klucze wykrywania związane z maszynami wirtualnymi.

Klucz pozycji		
Opis	**Pole**	**Pobrana zawartość**
vmware.cluster.discovery		
Wykonuje wykrywanie klastrów.	{#CLUSTER.ID}	Identyfikator klastra.
	{#CLUSTER.NAME}	Nazwa klastra.

Klucz pozycji		

	”resource_pool”	Tablica zawierająca dane puli zasobów, w tym identyfikator grupy zasobów, tablicę tagów, ścieżkę puli zasobów, liczbę maszyn wirtualnych.
		
		Struktura tablicy:
		`[{`
		`"rpid":"resource group id",`
		`"tags":[{}],`
		`"rpath":"resource group path",`
		`"vm_count":0`
		`}]`
		
		Strukturę tablicy `"tags"` opisano w polu `"tags"`.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.datastore.discovery		
Wykonuje wykrywanie datastore.	{#DATASTORE}	Nazwa datastore.

	{#DATASTORE.EXTENT}	Tablica zawierająca identyfikator partycji extent datastore i nazwę instancji.
		
		Struktura tablicy:
		`[{`
		`"partitionid":1,`
		`"instance":"name"`
		`}]`

	{#DATASTORE.TYPE}	Typ datastore. Przykładowe wartości: `VMFS`, `NFS`, `vsan` itd.
	{#DATASTORE.UUID}	Identyfikator datastore.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.dc.discovery		
Wykonuje wykrywanie centrów danych.	{#DATACENTER}	Nazwa centrum danych.
	{#DATACENTERID}	Identyfikator centrum danych.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.dvswitch.discovery		
Wykonuje wykrywanie rozproszonych przełączników vSphere.	{#DVS.NAME}	Nazwa przełącznika.

Klucz pozycji		
	{#DVS.UUID}	Identyfikator przełącznika.
vmware.hv.discovery		
Wykonuje wykrywanie hipernadzorców.	{#HV.UUID}	Unikalny identyfikator hipernadzorcy.
	{#HV.ID}	Identyfikator hipernadzorcy (nazwa obiektu zarządzanego HostSystem).
	{#HV.NAME}	Nazwa hipernadzorcy.
	{#HV.NETNAME}	Nazwa hosta sieciowego hipernadzorcy.
	{#HV.IP}	Adres IP hipernadzorcy, może być pusty. W przypadku konfiguracji HA z wieloma interfejsami sieciowymi obowiązuje następujący priorytet wyboru interfejsu: - preferuj adres IP, który współdzieli podsieć z adresem IP vCenter; - preferuj adres IP z podsieci z domyślną bramą; - preferuj adres IP z interfejsu o najniższym ID.
	{#CLUSTER.NAME}	Nazwa klastra, może być pusta.
	{#DATACENTER.NAME}	Nazwa centrum danych.
	{#PARENT.NAME}	Nazwa kontenera, w którym przechowywany jest hipernadzorca.
	{#PARENT.TYPE}	Typ kontenera, w którym przechowywany jest hipernadzorca. Wartości mogą być następujące: `Datacenter`, `Folder`, `ClusterComputeResource`, `VMware`, gdzie ”VMware” oznacza nieznany typ kontenera.

	”resource_pool”	Tablica zawierająca dane puli zasobów, w tym identyfikator grupy zasobów, tablicę tagów, ścieżkę puli zasobów, liczbę maszyn wirtualnych.
		
		Struktura tablicy:
		`[{`
		`"rpid":"resource group id",`
		`"tags":[{}],`
		`"rpath":"resource group path",`
		`"vm_count":0`
		`}]`
		
		Strukturę tablicy `"tags"` opisano w polu `"tags"`.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.hv.datastore.discovery		
Wykonuje wykrywanie datastore hipernadzorcy. Należy pamiętać, że wiele hipernadzorców może korzystać z tego samego datastore.	{#DATASTORE}	Nazwa datastore.
	{#DATASTORE.TYPE}	Typ datastore. Przykładowe wartości: `VMFS`, `NFS`, `vsan` itd.
	{#DATASTORE.UUID}	Identyfikator datastore.
	{#MULTIPATH.COUNT}	Zarejestrowana liczba ścieżek datastore.
	{#MULTIPATH.PARTITION.COUNT}	Liczba dostępnych partycji dyskowych.

Klucz pozycji		

	”datastore_extent”	Tablica zawierająca nazwę instancji extent datastore i identyfikator partycji.
		
		Struktura tablicy:
		`[{`
		`"partitionid":1,`
		`"instance":"name"`
		`}]`

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

vmware.hv.net.if.discovery		
Wykonuje wykrywanie interfejsów sieciowych hipernadzorcy.	{#IFNAME}	Nazwa interfejsu.
	{#IFDRIVER}	Sterownik interfejsu.
	{#IFDUPLEX}	Ustawienia dupleksu interfejsu.
	{#IFSPEED}	Prędkość interfejsu.
	{#IFMAC}	Adres MAC interfejsu.
vmware.vm.discovery		
Wykonuje wykrywanie maszyn wirtualnych.	{#VM.UUID}	Unikalny identyfikator maszyny wirtualnej.
	{#VM.ID}	Identyfikator maszyny wirtualnej (nazwa obiektu zarządzanego VirtualMachine).
	{#VM.NAME}	Nazwa maszyny wirtualnej.
	{#HV.NAME}	Nazwa hipernadzorcy.
	{#HV.UUID}	Unikalny identyfikator hipernadzorcy.
	{#HV.ID}	Identyfikator hipernadzorcy (nazwa obiektu zarządzanego HostSystem).
	{#CLUSTER.NAME}	Nazwa klastra, może być pusta.
	{#DATACENTER.NAME}	Nazwa centrum danych.
	{#DATASTORE.NAME}	Nazwa datastore.
	{#DATASTORE.UUID}	Identyfikator datastore.
	{#VM.IP}	Adres IP maszyny wirtualnej, może być pusty.
	{#VM.DNS}	Nazwa DNS maszyny wirtualnej, może być pusta.
	{#VM.GUESTFAMILY}	Rodzina systemu operacyjnego gościa maszyny wirtualnej, może być pusta.
	{#VM.GUESTFULLNAME}	Pełna nazwa systemu operacyjnego gościa maszyny wirtualnej, może być pusta.
	{#VM.FOLDER}	Łańcuch folderów nadrzędnych maszyny wirtualnej, może być użyty jako wartość dla zagnieżdżonych grup; nazwy folderów są łączone znakiem ”/”. Może być pusty.
	{#VM.TOOLS.STATUS}	Stan narzędzi maszyny wirtualnej VMware.
	{#VM.POWERSTATE}	Stan zasilania maszyny wirtualnej VMware (`poweredOff`, `poweredOn` lub `suspended`).
	{#VM.RPOOL.ID}	Identyfikator puli zasobów.
	{#VM.RPOOL.PATH}	Pełna ścieżka puli zasobów z wyłączeniem nazwy głównej ”Resources”. Nazwy folderów są łączone znakiem ”/”.
	{#VM.SNAPSHOT.COUNT}	Liczba migawek VM.

	”tags”	Tablica zawierająca tagi z nazwą tagu, opisem i kategorią.
		
		Struktura tablicy:
		`[{`
		`"tag":"tag name",`
		`"tag_description":"tag description",`
		`"category":"tag category"`
		`}]`

Klucz pozycji		

	”vm_customattribute”	Tablica niestandardowych atrybutów maszyny wirtualnej (jeśli zdefiniowane).
		
		Struktura tablicy:
		`[{`
		`"name":"custom field name",`
		`"value":"custom field value"`
		`}]`

	”net_if”	Tablica interfejsów sieciowych maszyny wirtualnej.
		
		Struktura tablicy:
		`[{`
		`"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"]`
		`}]`
		
		Opis zwracanych danych znajduje się w kluczu pozycji ”vmware.vm.net.if.discovery”.

vmware.vm.net.if.discovery		
Wykonuje wykrywanie interfejsów sieciowych maszyny wirtualnej.	{#IFNAME}	Nazwa interfejsu sieciowego.
	{#IFDESC}	Opis interfejsu.
	{#IFMAC}	Adres MAC interfejsu.
	{#IFCONNECTED}	Stan połączenia interfejsu (`false` - rozłączony; `true` - połączony).
	{#IFTYPE}	Typ interfejsu.
	{#IFBACKINGDEVICE}	Nazwa urządzenia bazowego.
	{#IFDVSWITCH.UUID}	Unikalny identyfikator vSphere Distributed Switch.
	{#IFDVSWITCH.PORTGROUP}	Rozproszona grupa portów.
	{#IFDVSWITCH.PORT}	Port vSphere Distributed Switch.
	”ifip”	Tablica adresów interfejsu.
vmware.vm.vfs.dev.discovery		
Wykonuje wykrywanie urządzeń dyskowych maszyny wirtualnej.	{#DISKNAME}	Nazwa urządzenia dyskowego.
vmware.vm.vfs.fs.discovery		
Wykonuje wykrywanie systemów plików maszyny wirtualnej.	{#FSNAME}	Nazwa systemu plików.

Przykłady JSON dla pozycji VMware

Przegląd Ta sekcja zawiera dodatkowe informacje o obiektach JSON zwracanych przez różne pozycje VMware.

vmware.*.alarms.get Pozycje **vmware.alarms.get[]**, **vmware.cluster.alarms.get[]**, **vmware.datastore.alarms.get[]**, **vmware.dc.alarms.get[]**, **vmware.hv.alarms.get[]**, **vmware.vm.alarms.get[]** zwracają obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "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 Pozycje **vmware.cluster.tags.get[]**, **vmware.datastore.tags.get[]**, **vmware.dc.tags.get[]**, **vmware.hv.tags.g**
vmware.vm.tags.get[] zwracają obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "tags": [
        {
            "name": "Windows",
        "description": "tag dla kategorii typ systemu operacyjnego",
        "category": "typ systemu operacyjnego"
    },
    {
            "name": "SQL Server",
        "description": "tag dla kategorii nazwa aplikacji",
        "category": "nazwa aplikacji"
        },
    {
            // inne tagi
        }
    ]
}
```

vmware.hv.diskinfo.get Pozycja **vmware.hv.diskinfo.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
[
  {
    "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": {}
  },
  {
    // other instances
  }
]
```

vmware.dvswitch.fetchports.get Pozycja **vmware.dvswitch.fetchports.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "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"
            },
            {
                //inne klucze
            }
        ]
    }
}
```

vmware.hv.hw.sensors.get Pozycja **vmware.hv.hw.sensors.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "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"
                 }
            },
            {
```

```
                    // inne czujniki sprzętowe hv
                }
        ]
        }
}
```

vmware.hv.sensors.get Pozycja **vmware.hv.sensors.get[]** zwraca obiekty JSON o następującej strukturze (wartości podano jako przykład):

```
{
    "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"
        },
                {
                    // inne sensory hv
                }
        ]
        }
}
```

vmware.vm.snapshot.get Jeśli istnieją jakiekolwiek snapshoty, pozycja **vmware.snapshot.get[]** zwraca obiekt JSON o następującej strukturze (wartości podano jako przykład):

```
{
  "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
}
```


Jeśli nie istnieje żaden snapshot, pozycja **vmware.snapshot.get[]** zwraca obiekt JSON z pustymi wartościami:


```
{
  "snapshot": [],
  "count": 0,
  "latestdate": null,
  "latestage": 0,
  "oldestdate": null,
  "oldestage": 0,
  "size": 0,
  "uniquesize": 0
}
```


Przykład konfiguracji monitorowania VMware

Przegląd

Poniższy przykład opisuje, jak skonfigurować Zabbix do monitorowania maszyn wirtualnych VMware. Obejmuje to:

- utworzenie hosta reprezentującego środowisko VMware;
- utworzenie reguły wykrywania niskiego poziomu, która wykrywa maszyny wirtualne w środowisku VMware;
- utworzenie prototypu hosta, na podstawie którego Zabbix wygeneruje rzeczywiste hosty dla maszyn wirtualnych wykrytych przez regułę wykrywania niskiego poziomu.

Wymagania wstępne

> **Note:**
> Ten przykład nie obejmuje konfiguracji VMware. Zakłada się, że VMware jest już skonfigurowane.

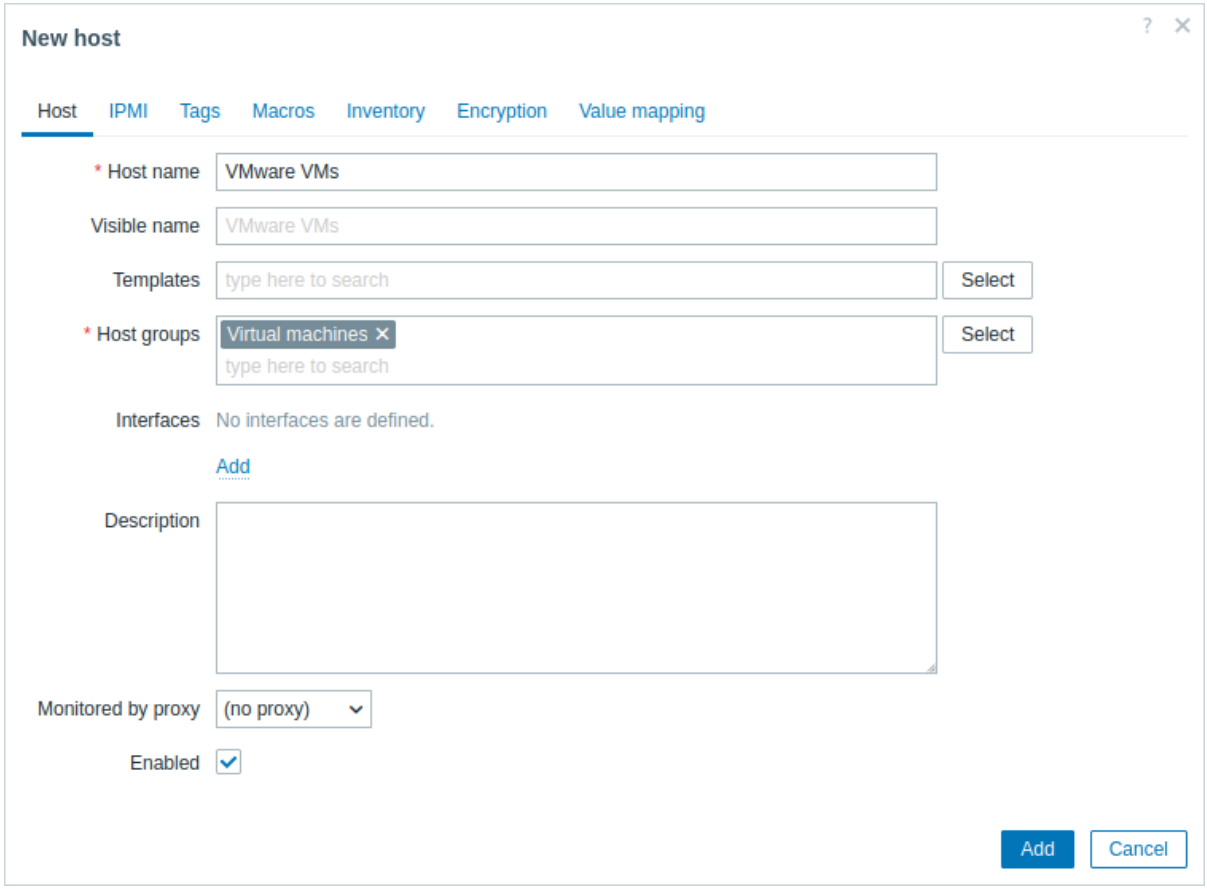
Przed kontynuowaniem ustaw parametr `StartVMwareCollectors` w pliku konfiguracyjnym serwer Zabbix na 2 lub więcej (wartość domyślna to 0).

Utwórz host

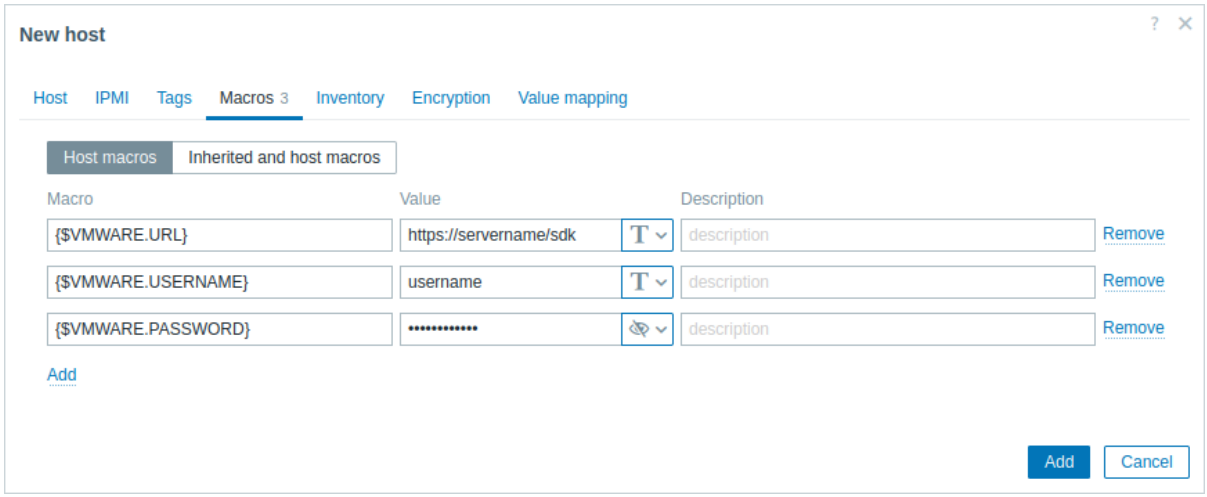
1. Przejdź do *Zbieranie danych* > *Hosty*.

2. Utwórz host:

- W polu *Nazwa hosta* wpisz nazwę hosta (na przykład ”VMware VMs”).
- W polu *Grupy hostów* wpisz lub wybierz grupę hostów (na przykład ”Virtual machines”).



- Na karcie *Makra* ustaw następujące makra hosta:
  - {$VMWARE.URL} - adres URL SDK usługi VMware (hiperwizor ESXi) (https://servername/sdk)
  - {$VMWARE.USERNAME} - nazwa użytkownika usługi VMware
  - {$VMWARE.PASSWORD} - hasło użytkownika {$VMWARE.USERNAME} usługi VMware



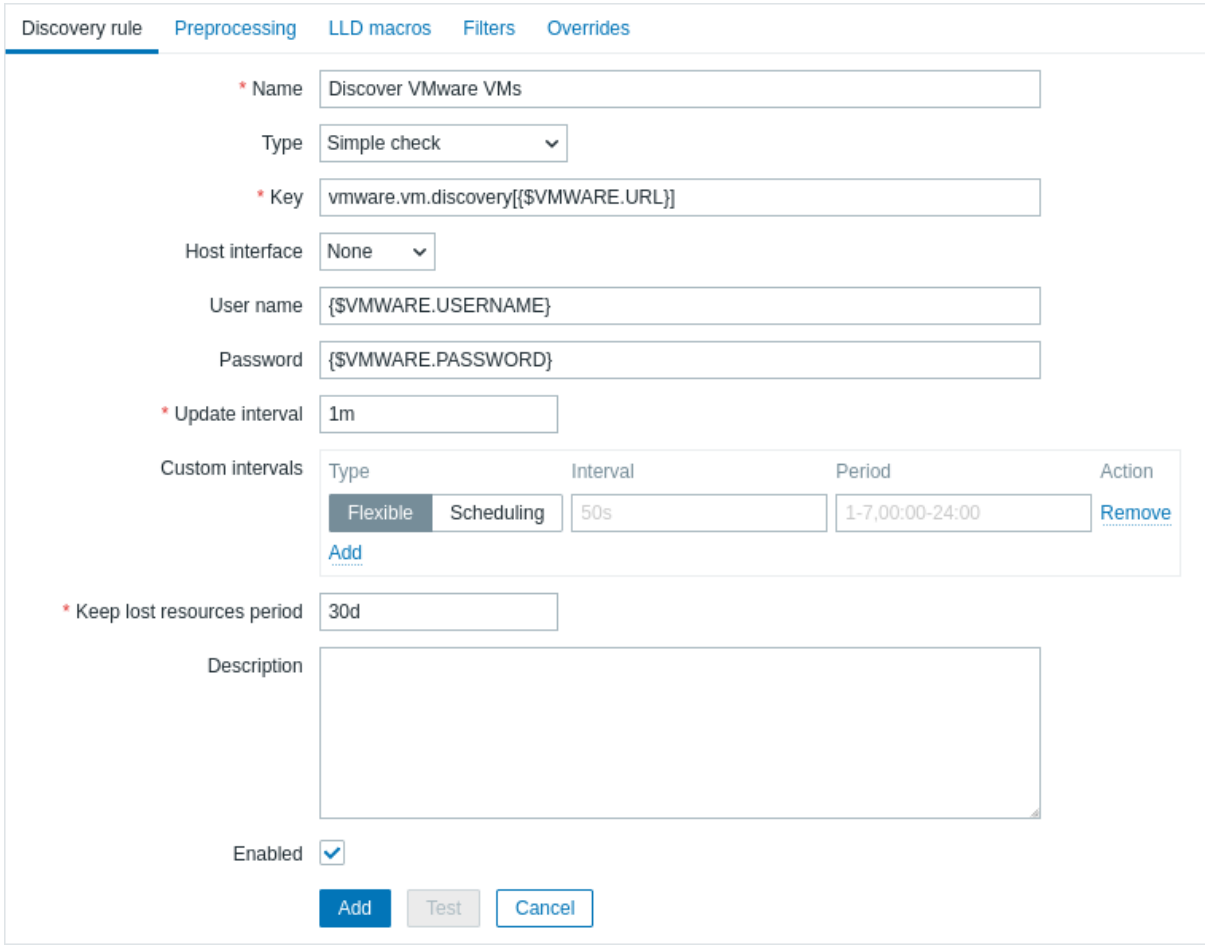
3. Kliknij przycisk *Dodaj*, aby utworzyć host. Ten host będzie reprezentować Twoje środowisko VMware.

Utwórz regułę wykrywania niskiego poziomu

1. Kliknij *Discovery* dla utworzonego hosta, aby przejść do listy reguł wykrywania niskiego poziomu dla tego hosta.

2. Utwórz regułę wykrywania niskiego poziomu:

- W polu *Name* wprowadź nazwę reguły wykrywania niskiego poziomu (na przykład „Wykryj maszyny wirtualne VMware”).
- W polu *Type* wybierz „Simple check”.
- W polu *Key* wprowadź wbudowany klucz pozycji do wykrywania maszyn wirtualnych VMware: `vmware.vm.discovery[{$VMWARE.URL`
- W polach *User name* i *Password* wprowadź odpowiednie makra wcześniej skonfigurowane na hoście.



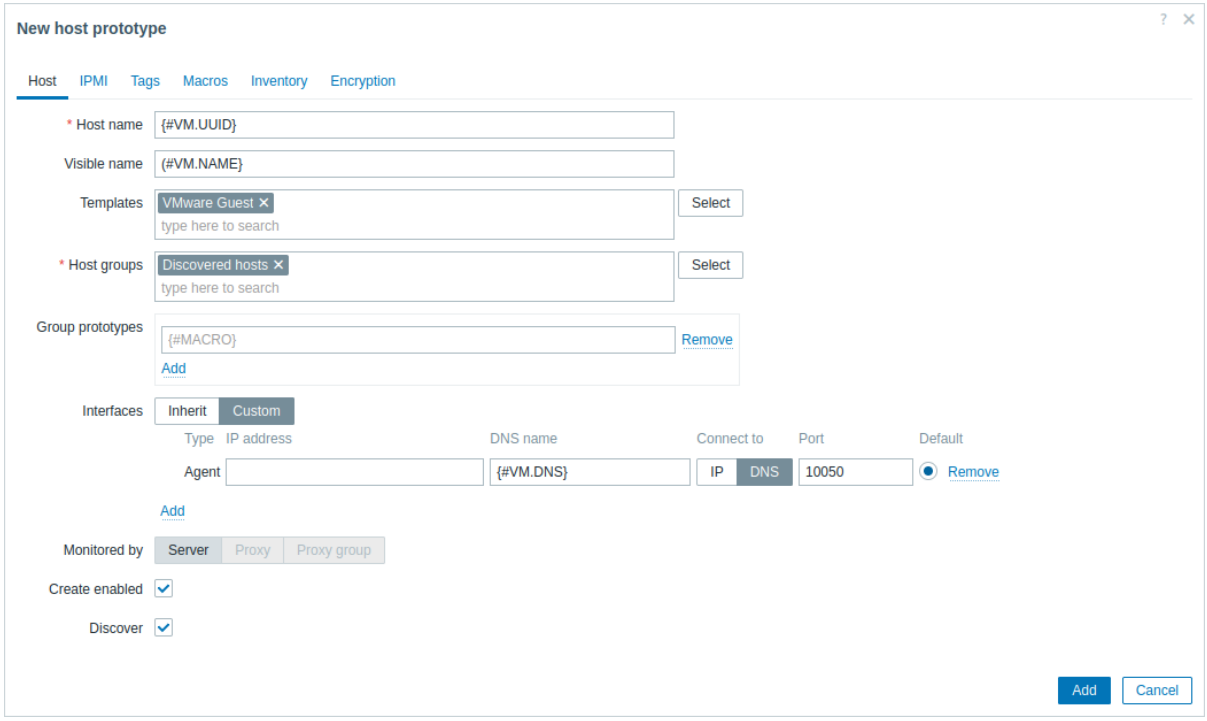
3. Kliknij przycisk *Add*, aby utworzyć regułę wykrywania niskiego poziomu. Ta reguła wykrywania wykryje maszyny wirtualne w Twoim środowisku VMware.

Utwórz prototyp hosta

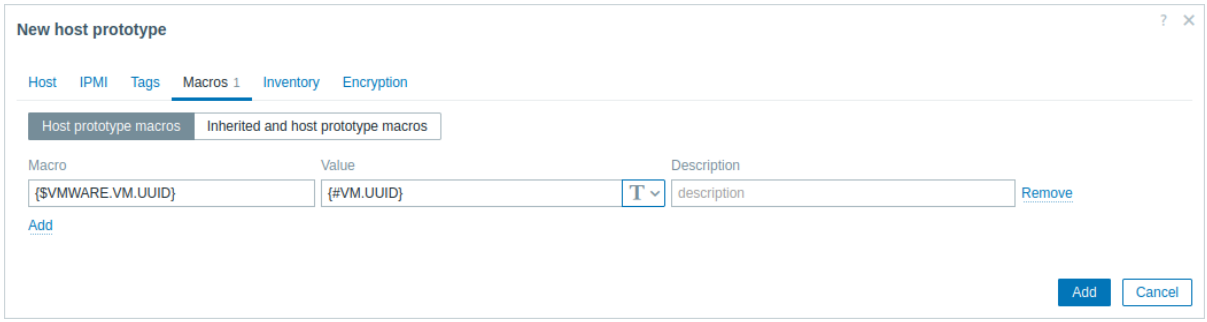
1. Na liście reguł wykrywania niskiego poziomu kliknij *Host prototypes* dla wcześniej utworzonej reguły wykrywania niskiego poziomu.

2. Utwórz prototyp hosta. Ponieważ prototypy hostów są wzorcami do tworzenia hostów za pomocą reguł wykrywania niskiego poziomu, większość pól będzie zawierać makra wykrywania niskiego poziomu. Zapewnia to tworzenie hostów z właściwościami opartymi na zawartości pobranej przez wcześniej utworzoną regułę wykrywania niskiego poziomu.

- W polu *Host name* wprowadź makro {#VM.UUID}.
- W polu *Visible name* wprowadź makro {#VM.NAME}.
- W polu *Templates* wprowadź lub wybierz szablon ”VMware Guest”. Ten szablon zawiera pozycje VMware oraz reguły wykrywania do monitorowania stanu zasilania maszyny wirtualnej, użycia CPU, użycia pamięci, urządzeń sieciowych itp.
- W polu *Host groups* wprowadź lub wybierz grupę hostów (na przykład ”Discovered hosts”).
- W polu *Interfaces* dodaj niestandardowy interfejs hosta. Następnie wprowadź makro {#VM.DNS} w polu *DNS name* lub wprowadź makro {#VM.IP} w polu *IP address*. Alternatywnie, jeśli maszyny wirtualne w Twoim środowisku VMware mają wiele interfejsów, przejdź do sekcji *Zaawansowana konfiguracja interfejsu hosta*. Skonfigurowanie niestandardowego interfejsu hosta jest konieczne do prawidłowego działania szablonu *VMware Guest*.



- Na karcie *Macros* ustaw makro {$VMWARE.VM.UUID} z wartością {#VM.UUID}. Jest to konieczne do prawidłowego działania szablonu *VMware Guest*, który używa tego makra jako makra użytkownika na poziomie hosta w parametrach pozycji (na przykład *vmware.vm.net.if.discovery[{$VMWARE.URL}, {$VMWARE.VM.UUID}]*).

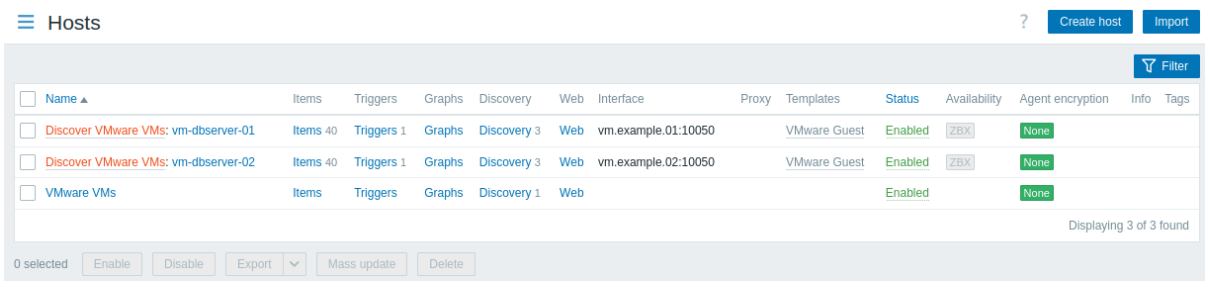


3. Kliknij przycisk *Add*, aby utworzyć prototyp hosta. Ten prototyp hosta będzie używany do tworzenia hostów dla maszyn wirtualnych wykrytych przez wcześniej utworzoną regułę wykrywania niskiego poziomu.

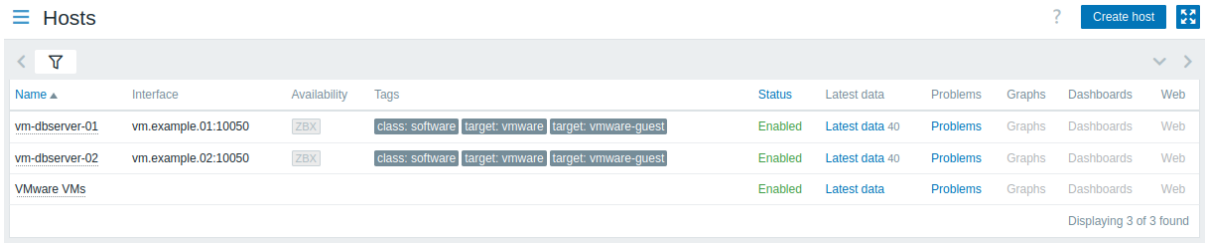
Wyświetlanie hostów i metryk

Po utworzeniu prototypu hosta reguła wykrywania niskiego poziomu utworzy hosty dla wykrytych maszyn wirtualnych VMware, a Zabbix rozpocznie ich monitorowanie. Należy pamiętać, że wykrywanie i tworzenie hostów może zostać również wykonane ręcznie, jeśli zajdzie taka potrzeba.

Aby wyświetlić utworzone hosty, przejdź do sekcji menu *Zbieranie danych > Hosty*.



Aby wyświetlić zebrane metryki, przejdź do sekcji menu *Monitoring > Hosty* i kliknij *Najnowsze dane* dla jednego z hostów.



Zaawansowana konfiguracja interfejsu hosta

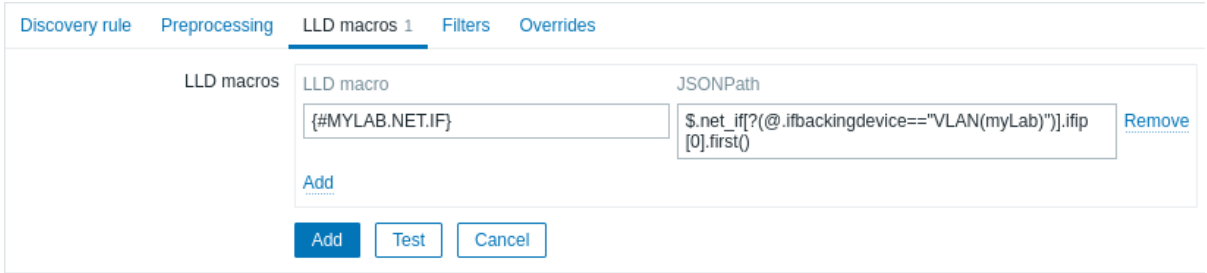
Klucz pozycji `vmware.vm.discovery[{$VMWARE.URL}]`, skonfigurowany w sekcji *Utwórz regułę wykrywania niskiego poziomu*, zwraca dane interfejsów sieciowych w polu "net_if":

```
"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": []
      }
    ]
```

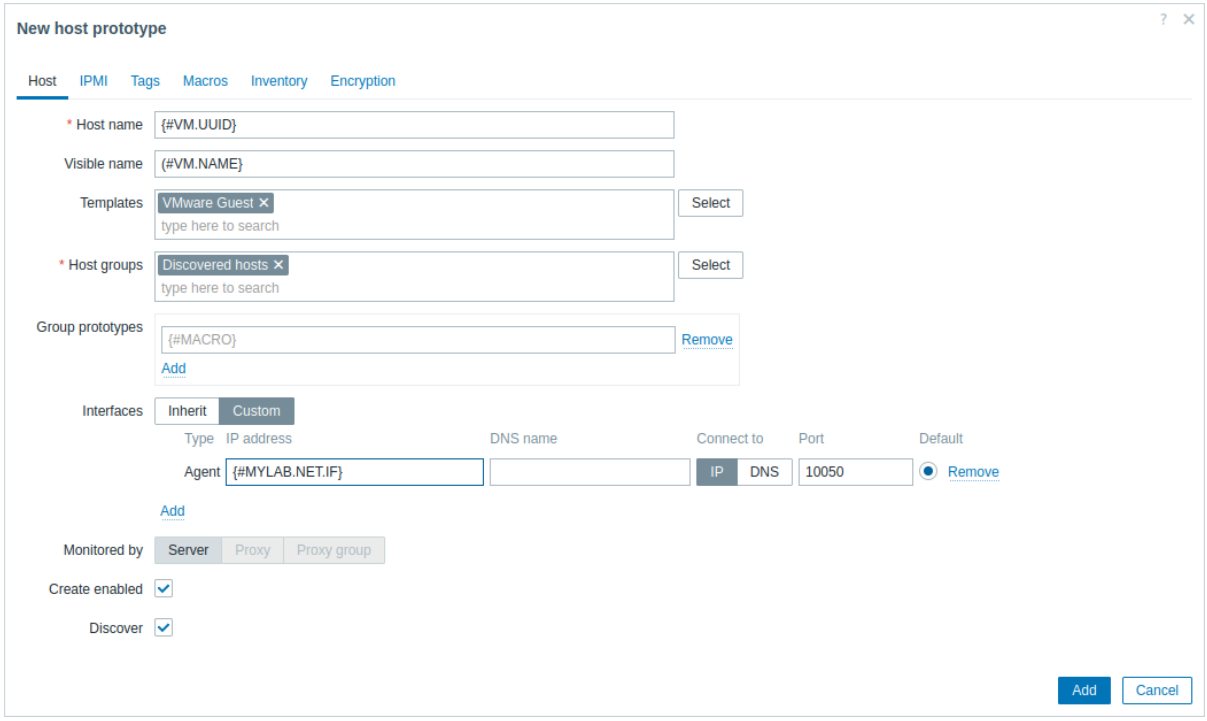
Dane te można wykorzystać do skonfigurowania niestandardowego interfejsu hosta.

1. Podczas tworzenia reguły wykrywania niskiego poziomu dodatkowo skonfiguruj makro wykrywania niskiego poziomu. Na karcie *Makra LLD* utwórz niestandardowe makro LLD z wartością JSONPath. Na przykład:

- `{#MYLAB.NET.IF}` - `$.net_if[?(@.ifbackingdevice=="VLAN(myLab)")].ifip[0].first()`



2. Podczas tworzenia prototypu hosta dodaj niestandardowy interfejs hosta i wprowadź makro LLD w polu *Nazwa DNS* lub *Adres IP*.



Monitorowanie sieci web

Przegląd Za pomocą Zabbix możesz sprawdzać kilka aspektów dostępności witryn internetowych.

Attention:
Aby wykonywać monitorowanie WWW, serwer Zabbix musi być początkowo skonfigurowany z obsługą cURL (libcurl).

Aby aktywować monitorowanie WWW, musisz zdefiniować scenariusze WWW. Scenariusz WWW składa się z jednego lub kilku żądań HTTP albo ”kroków”. Kroki są okresowo wykonywane przez serwer Zabbix w z góry określonej kolejności. Jeśli host jest monitorowany przez proxy, kroki są wykonywane przez proxy.

Scenariusze WWW są przypisywane do hostów/szablonów w taki sam sposób jak pozycje, wyzwalacze itp. Oznacza to, że scenariusze WWW można również tworzyć na poziomie szablonu, a następnie zastosować do wielu hostów jednym działaniem.

W każdym scenariuszu WWW zbierane są następujące informacje:

- średnia prędkość pobierania na sekundę dla wszystkich kroków całego scenariusza
- numer kroku, który zakończył się niepowodzeniem
- ostatni komunikat o błędzie

W każdym kroku scenariusza WWW zbierane są następujące informacje:

- prędkość pobierania na sekundę
- czas odpowiedzi
- kod odpowiedzi

Więcej informacji znajdziesz w sekcji pozycje monitorowania WWW.

Dane zbierane podczas wykonywania scenariuszy WWW są przechowywane w bazie danych. Dane są automatycznie wykorzystywane do wykresów, wyzwalaczy i powiadomień.

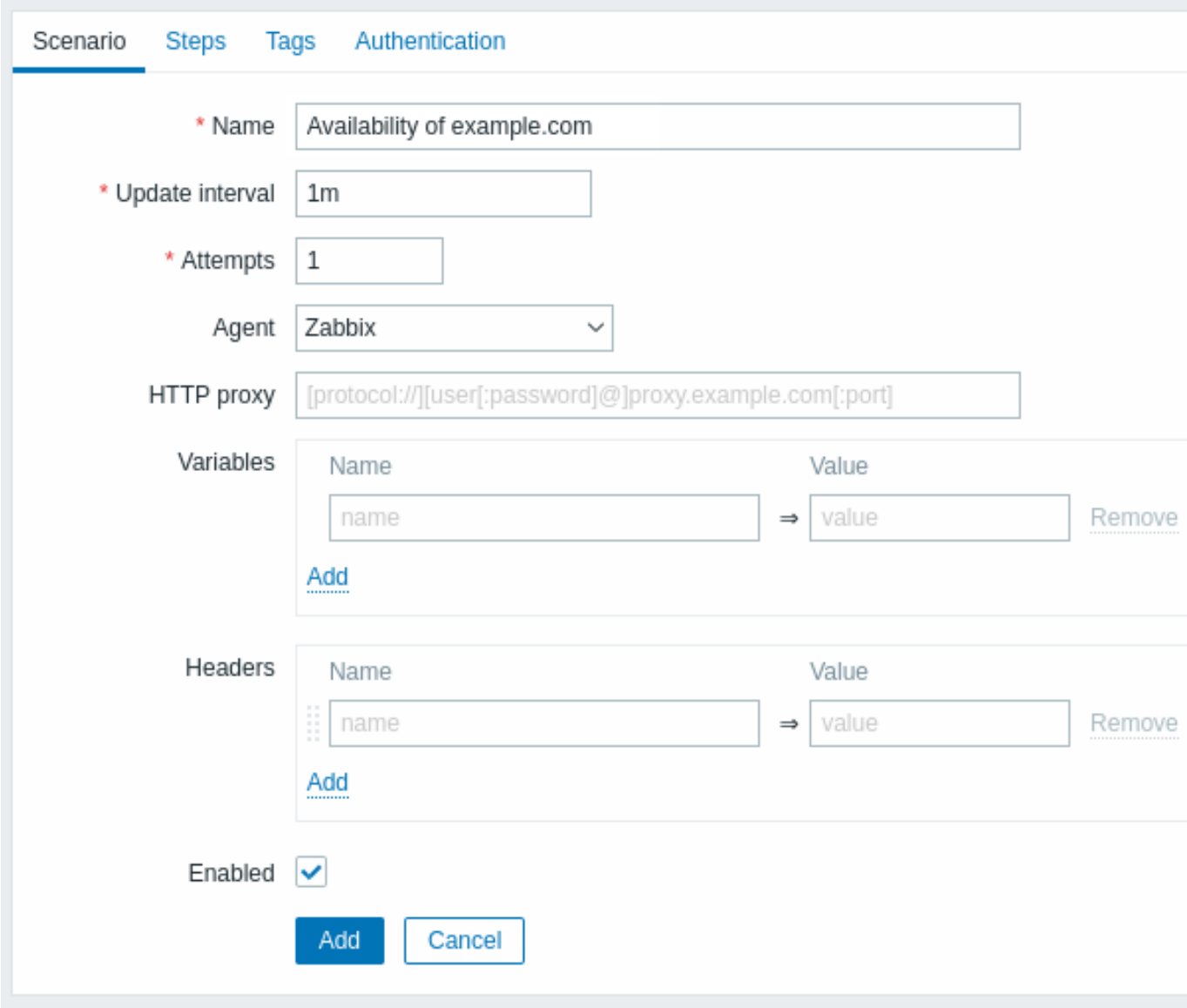
Zabbix może również sprawdzać, czy pobrana strona HTML zawiera z góry określony ciąg znaków. Może wykonać symulowane logowanie i przejść przez ścieżkę symulowanych kliknięć myszą na stronie.

Monitorowanie WWW w Zabbix obsługuje zarówno HTTP, jak i HTTPS. Podczas uruchamiania scenariusza WWW Zabbix opcjonalnie będzie podążać za przekierowaniami (zobacz opcję *Follow redirects* poniżej). Maksymalna liczba przekierowań jest na sztywno ustawiona na 10 (z użyciem opcji cURL CURLOPT_MAXREDIRS). Wszystkie pliki cookie są zachowywane podczas wykonywania pojedynczego scenariusza.

Konfigurowanie scenariusza web Aby skonfigurować scenariusz web:

1. Przejdź do: *Gromadzenie danych* > *Hosty* (lub *Szablony*).
2. Kliknij *Web* w wierszu hosta/szablonu.
3. Kliknij po prawej stronie *Utwórz scenariusz web* (lub nazwę scenariusza, aby edytować istniejący scenariusz).
4. Wprowadź parametry scenariusza w formularzu.

Karta **Scenariusz** umożliwia skonfigurowanie ogólnych parametrów scenariusza web.



Wszystkie wymagane pola wejściowe są oznaczone czerwoną gwiazdką.

Parametry scenariusza:

Parametr	Opis
Nazwa	Unikalna nazwa scenariusza. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane są makra użytkownika, pozostaną one nierozwinięte w nazwach pozycji monitorowania web.
Interwał aktualizacji	Jak często scenariusz będzie wykonywany. Obsługiwane są przyrostki czasu, np. 30s, 1m, 2h, 1d. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane jest makro użytkownika, a jego wartość zostanie zmieniona (np. 5m → 30s), następne sprawdzenie zostanie wykonane zgodnie z poprzednią wartością (w przykładzie z podanymi wartościami nastąpi później). Nowe scenariusze web zostaną sprawdzone w ciągu 60 sekund od ich utworzenia.

Parametr	Opis
Próby	Liczba prób wykonania kroków scenariusza web. W przypadku problemów z siecią (przekroczenie limitu czasu, brak połączenia itp.) Zabbix może kilkakrotnie powtórzyć wykonanie kroku. Ustawiona wartość będzie jednakowo dotyczyć każdego kroku scenariusza. Można określić maksymalnie 10 prób; wartość domyślna to 1. *Uwaga*: Zabbix nie powtórzy kroku z powodu nieprawidłowego kodu odpowiedzi ani niezgodności wymaganego ciągu znaków.
Agent	Wybierz agenta klienta. Zabbix będzie symulować wybraną przeglądarkę. Jest to przydatne, gdy witryna zwraca inną zawartość dla różnych przeglądarek. W tym polu można używać makr użytkownika.
Proxy HTTP	Można określić proxy HTTP do użycia, korzystając z formatu `[protocol://][username[:password]@]proxy.example.com[:port]`. Ustawia to opcję cURL CURLOPT_PROXY. Opcjonalny prefiks `protocol://` może służyć do określenia alternatywnych protokołów proxy (obsługa prefiksu protokołu została dodana w cURL 7.21.7). Jeśli protokół nie zostanie określony, proxy będzie traktowane jako proxy HTTP. Domyślnie używany będzie port 1080. Jeśli proxy zostanie określone, zastąpi powiązane z proxy zmienne środowiskowe, takie jak http_proxy i HTTPS_PROXY. Jeśli nie zostanie określone, nie zastąpi zmiennych środowiskowych związanych z proxy. Wprowadzona wartość jest przekazywana „bez zmian” i nie jest sprawdzana pod kątem poprawności. Można również wprowadzić adres proxy SOCKS. Jeśli zostanie określony nieprawidłowy protokół, połączenie nie powiedzie się, a pozycja stanie się nieobsługiwana. *Uwaga*: w przypadku proxy HTTP obsługiwane jest tylko proste uwierzytelnianie. W tym polu można używać makr użytkownika.
Zmienne	Zmienne, których można używać w krokach scenariusza (URL, zmienne post). Mają one następujący format: **{macro1}**=value1 **{macro2}**=value2 **{macro3}**=regex:<regular expression> **{macro4}**=jsonpath:<jsonpath> **{macro5}**=xmlxpath:<xmlxpath> **{macro6}**={{macro}.function()} (zobacz funkcje makr) Na przykład: {username}=Alexei {password}=kj3h5kJ34bd {hostid}=regex:hostid is ([0-9]+) {url}=jsonpath:$.host_url {status}=xmlxpath://host/response/status {newvar}={{myvar}.btoa()} Następnie można odwoływać się do makr w krokach jako {username}, {password}, {hostid} itd. Zabbix automatycznie zastąpi je rzeczywistymi wartościami. Należy pamiętać, że zmienne z `regex:` wymagają jednego kroku pobierającego wartość wyrażenia regularnego, dlatego wyodrębniona wartość może zostać zastosowana dopiero w następnym kroku. Jeśli część wartości zaczyna się od `regex:`, część znajdująca się za nim jest traktowana jako wyrażenie regularne wyszukujące dane na stronie internetowej i, jeśli zostanie znalezione, zapisujące dopasowanie w zmiennej. Musi być obecna co najmniej jedna podgrupa, aby można było wyodrębnić dopasowaną wartość. Obsługiwane są makra użytkownika oraz makra `{HOST.*}`. Zmienne są automatycznie kodowane w formacie URL, gdy są używane w polach zapytania lub danych formularza dla zmiennych post, ale podczas używania w surowym poście lub bezpośrednio w URL muszą być kodowane w formacie URL ręcznie.

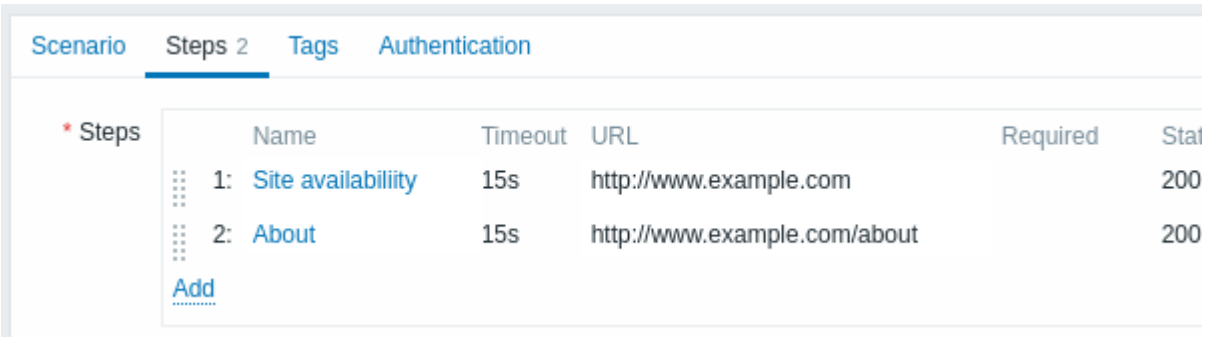
Parametr	Opis
Nagłówki	Nagłówki HTTP są używane podczas wykonywania żądania. Można używać nagłówków domyślnych i niestandardowych. Nagłówki zostaną przypisane przy użyciu ustawień domyślnych zależnych od typu agenta wybranego z listy rozwijanej na poziomie scenariusza i zastosowane do wszystkich kroków, chyba że zostaną niestandardowo zdefiniowane na poziomie kroku. **Należy pamiętać, że zdefiniowanie nagłówka na poziomie kroku automatycznie odrzuca wszystkie wcześniej zdefiniowane nagłówki, z wyjątkiem domyślnego nagłówka przypisanego przez wybranie „User-Agent” z listy rozwijanej na poziomie scenariusza.** Jednak nawet domyślny nagłówek „User-Agent” może zostać zastąpiony przez określenie go na poziomie kroku. Aby usunąć nagłówek na poziomie scenariusza, należy nadać mu nazwę i pozostawić jego wartość pustą na poziomie kroku. Nagłówki należy wyświetlać przy użyciu tej samej składni, w jakiej występują w protokole HTTP, opcjonalnie korzystając z dodatkowych funkcji obsługiwanych przez opcję cURL CURLOPT_HTTPHEADER. Na przykład: Accept-Charset=utf-8 Accept-Language=en-US Content-Type=application/xml; charset=utf-8 Obsługiwane są makra użytkownika oraz makra `{HOST.*}`.
Włączony	Scenariusz jest aktywny, jeśli to pole wyboru jest zaznaczone; w przeciwnym razie jest wyłączony.

Należy pamiętać, że podczas edytowania istniejącego scenariusza w formularzu dostępne są dwa dodatkowe przyciski:

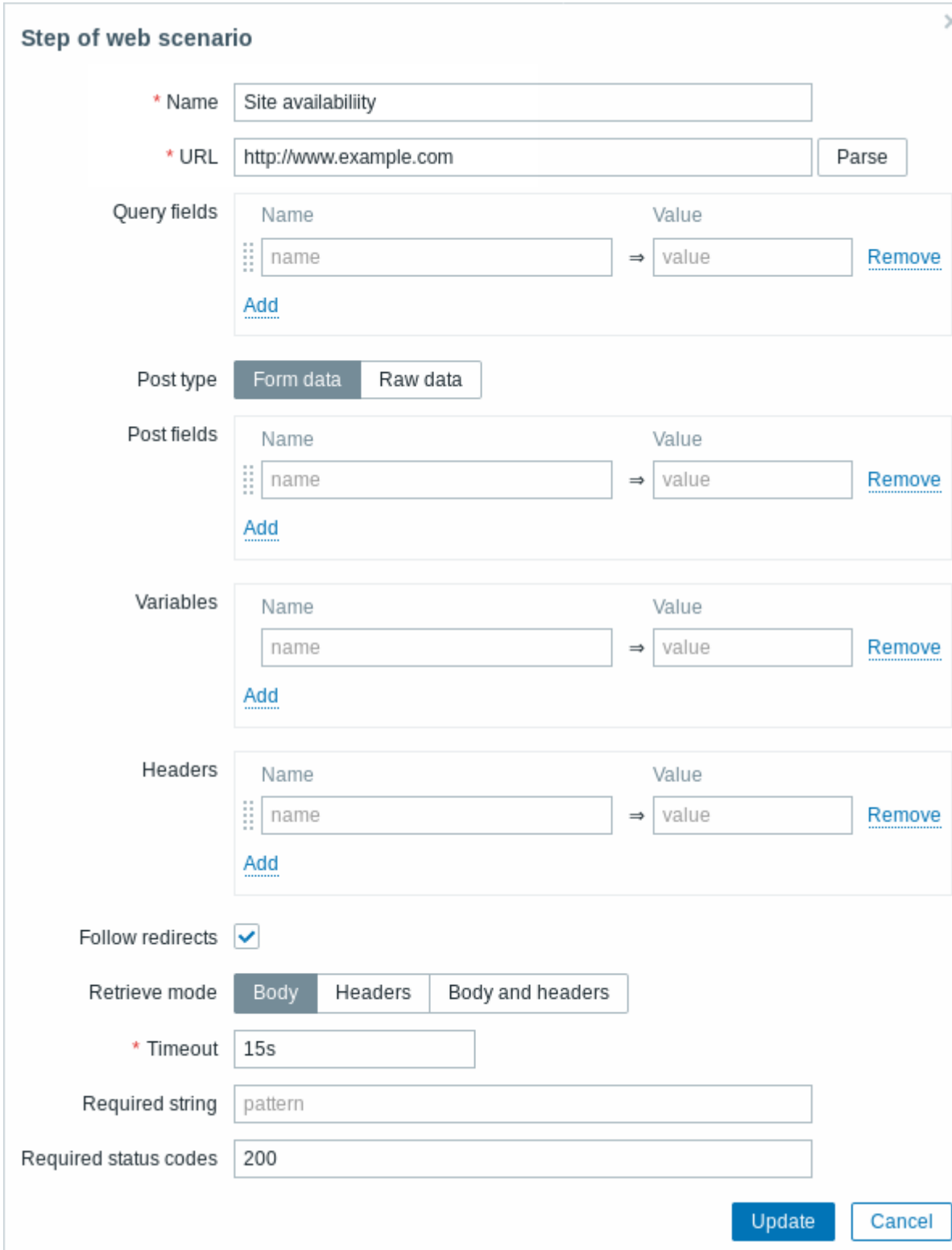
Clone	Utworzenie kolejnego scenariusza na podstawie właściwości istniejącego scenariusza.
Clear history and trends	Usunięcie danych historycznych i danych trendów dla scenariusza. Spowoduje to, że serwer wykona scenariusz natychmiast po usunięciu danych.

Note:
Jeśli pole *Proxy HTTP* pozostanie puste, innym sposobem użycia proxy HTTP jest ustawienie zmiennych środowiskowych związanych z proxy.
W przypadku sprawdzeń HTTP ustaw zmienną środowiskową `http_proxy` dla użytkownika serwera Zabbix. Na przykład: `http_proxy=http://proxy_ip:proxy_port`.
W przypadku sprawdzeń HTTPS ustaw zmienną środowiskową `HTTPS_PROXY`. Na przykład: `HTTPS_PROXY=http://proxy_ip:proxy_port`. Więcej informacji można uzyskać, uruchamiając polecenie powłoki: `# man curl`.

Karta **Kroki** umożliwia skonfigurowanie kroków scenariusza web. Aby dodać krok scenariusza web, kliknij *Dodaj* w bloku *Kroki*.



Note:
Tajnych makr użytkownika nie wolno używać w adresach URL, ponieważ zostaną rozwinięte do wartości ”******”.



Konfigurowanie kroków

Parametry kroku:

Parametr	Opis
Nazwa	Unikalna nazwa kroku. Obsługiwane są makra użytkownika. *Uwaga*: jeśli używane są makra użytkownika, pozostaną one nierozwinięte w nazwach pozycji monitorowania sieci WWW.

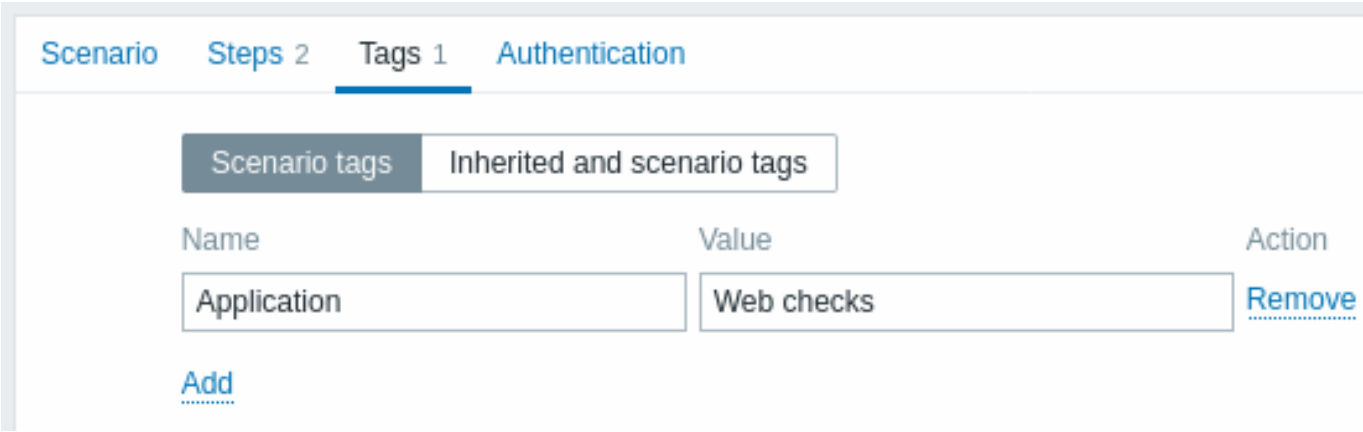
Parametr	Opis
URL	URL, z którym należy się połączyć i z którego należy pobrać dane. Na przykład: https://www.example.com http://www.example.com/download Nazwy domen mogą być określane za pomocą znaków Unicode. Podczas wykonywania kroku scenariusza sieci WWW są one automatycznie konwertowane na ASCII w formacie punycode. Przycisk *Parse* umożliwia oddzielenie opcjonalnych pól zapytania (takich jak `?key=value&limit=10`) od URL, przenosząc atrybuty i wartości do pola *Query fields* w celu automatycznego kodowania URL. W URL można używać zmiennych, stosując składnię {macro}. Zmienne można ręcznie kodować w URL za pomocą składni {{macro}.urlencode()}. Obsługiwane są makra użytkownika i makra `{HOST.*}` makra. Ze względów bezpieczeństwa nie należy umieszczać rzeczywistych nazw użytkowników ani haseł w URL. Zamiast tego należy używać makr użytkownika, aby uniknąć ujawnienia danych uwierzytelniających. Ograniczenie do 2048 znaków.
Query fields	Zmienne HTTP GET dla URL. Określane jako pary atrybutów i wartości. Wartości są automatycznie kodowane w URL. Wartości ze zmiennych scenariusza, makr użytkownika lub makr `{HOST.*}` są rozwijane, a następnie automatycznie kodowane w URL. Użycie składni {{macro}.urlencode()} spowoduje ich podwójne kodowanie w URL. Obsługiwane są makra użytkownika i makra `{HOST.*}` makra.
Post	Zmienne HTTP POST. W trybie **Form data** określane jako pary atrybutów i wartości. Wartości są automatycznie kodowane w URL. Wartości ze zmiennych scenariusza, makr użytkownika lub makr `{HOST.*}` są rozwijane, a następnie automatycznie kodowane w URL. W trybie **Raw data** atrybuty/wartości są wyświetlane w jednym wierszu i łączone symbolem **&**. Wartości surowe można ręcznie kodować/dekodować w URL za pomocą składni {{macro}.urlencode()} lub {{macro}.urldecode()}. Na przykład: id=2345&userid={user} Jeśli {user} jest zdefiniowane jako zmienna scenariusza sieci WWW, zostanie zastąpione swoją wartością podczas wykonywania kroku. Aby zakodować zmienną w URL, należy zastąpić {user} przez {{user}.urlencode()}. Obsługiwane są makra użytkownika i makra `{HOST.*}` makra.
Variables	Zmienne na poziomie kroku, których można używać dla funkcji GET i POST. Określane jako pary atrybutów i wartości. Zmienne na poziomie kroku zastępują zmienne na poziomie scenariusza lub zmienne z poprzedniego kroku. Jednak wartość zmiennej na poziomie kroku wpływa dopiero na następny krok (a nie na bieżący krok). Mają następujący format: **{macro}**=value **{macro}**=regex:<regular expression> Więcej informacji zawiera opis zmiennych na poziomie scenariusza. Zmienne są automatycznie kodowane w URL, gdy są używane w polach zapytania lub danych formularza dla zmiennych post, ale w przypadku użycia w surowym poście lub bezpośrednio w URL należy zakodować je ręcznie.
Headers	Niestandardowe nagłówki HTTP wysyłane podczas wykonywania żądania. Określane jako pary atrybutów i wartości. Nagłówek zdefiniowany na poziomie kroku będzie używany dla danego kroku. **Należy pamiętać, że zdefiniowanie nagłówka na poziomie kroku automatycznie odrzuca wszystkie wcześniej zdefiniowane nagłówki, z wyjątkiem domyślnego nagłówka przypisanego przez wybranie „User-Agent” z listy rozwijanej na poziomie scenariusza.** Jednak nawet domyślny nagłówek „User-Agent” można zastąpić, określając go na poziomie kroku. Na przykład przypisanie nazwy nagłówkowi bez ustawienia wartości spowoduje usunięcie domyślnego nagłówka na poziomie scenariusza. Obsługiwane są makra użytkownika i makra `{HOST.*}`. Ustawia to opcję cURL `CURLOPT_HTTPHEADER`.
Follow redirects	Zaznacz pole wyboru, aby śledzić przekierowania HTTP. Ustawia to opcję cURL CURLOPT_FOLLOWLOCATION.

Parametr	Opis
Retrieve mode	Wybierz tryb pobierania: **Body** - pobieranie tylko treści z odpowiedzi HTTP **Headers** - pobieranie tylko nagłówków z odpowiedzi HTTP **Body and headers** - pobieranie treści i nagłówków z odpowiedzi HTTP
Timeout	Zabbix nie poświęci na przetwarzanie URL więcej czasu niż ustawiona wartość (od jednej sekundy do maksymalnie 1 godziny). W rzeczywistości ten parametr określa maksymalny czas nawiązania połączenia z URL oraz maksymalny czas wykonywania żądania HTTP. Dlatego Zabbix nie poświęci na krok więcej niż **2 x Timeout** sekund. Obsługiwane są przyrostki czasu, np. 30s, 1m, 1h. Obsługiwane są makra użytkownika.
Required string	Wymagany wzorzec wyrażenia regularnego. Jeśli pobrana treść (HTML) nie pasuje do wymaganego wzorca, krok zakończy się niepowodzeniem. Jeśli pole jest puste, sprawdzanie wymaganego ciągu nie jest wykonywane. Na przykład: Homepage of Zabbix Welcome.*admin *Uwaga*: Odwoływanie się w tym polu do wyrażeń regularnych utworzonych w frontend Zabbix nie jest obsługiwane. Obsługiwane są makra użytkownika i makra `{HOST.*}` makra.
Required status codes	Lista oczekiwanych kodów statusu HTTP. Jeśli Zabbix otrzyma kod, którego nie ma na liście, krok zakończy się niepowodzeniem. Jeśli pole jest puste, sprawdzanie kodów statusu nie jest wykonywane. Na przykład: 200,201,210-299 Obsługiwane są makra użytkownika.

Note:
Wszelkie zmiany w krokach scenariusza sieci WWW zostaną zapisane dopiero po zapisaniu całego scenariusza.

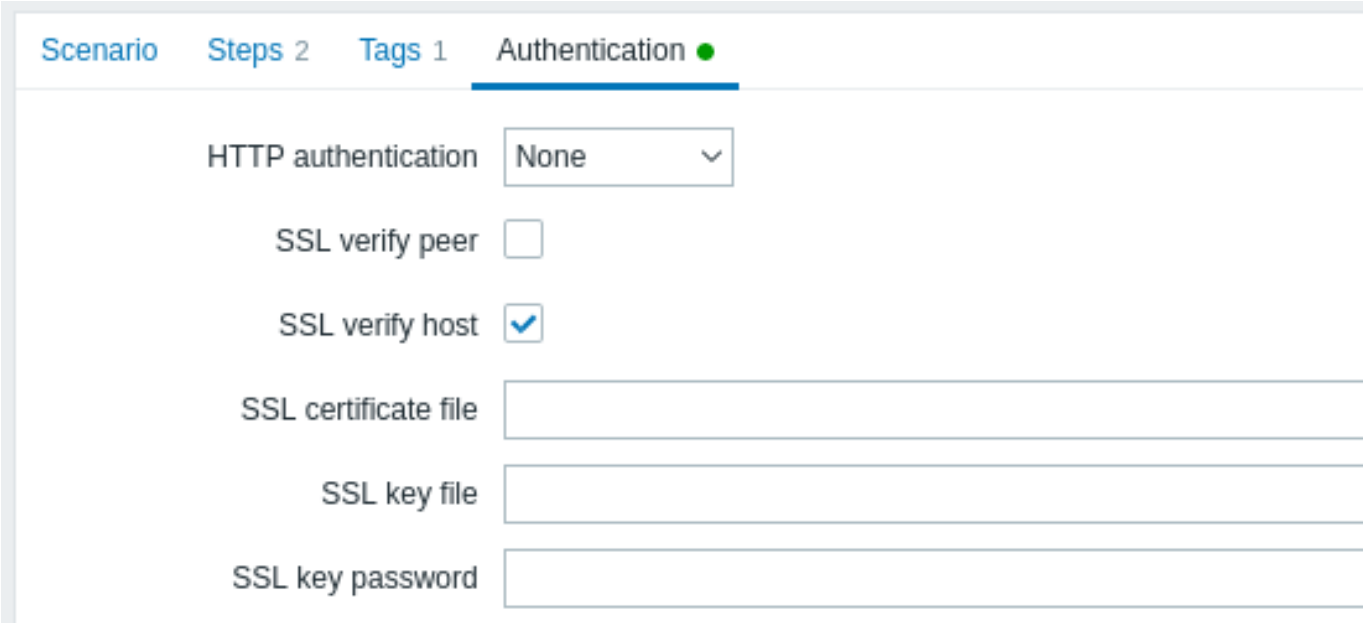
Zobacz także przykład z życia wzięty pokazujący, jak można skonfigurować kroki monitorowania sieci WWW.

Konfigurowanie tagów Karta **Tagi** umożliwia definiowanie tagów na poziomie scenariusza.



Tagowanie umożliwia filtrowanie scenariuszy webowych i pozycji monitorowania webowego.

Konfigurowanie uwierzytelniania Karta **Uwierzytelnianie** umożliwia skonfigurowanie opcji uwierzytelniania scenariusza. Zielona kropka obok nazwy karty wskazuje, że włączono pewien typ uwierzytelniania HTTP.



Parametry uwierzytelniania:

Parametr	Opis
Uwierzytelnianie HTTP	Wybierz opcję uwierzytelniania: **Brak** - uwierzytelnianie nie jest używane; **Basic** - używane jest uwierzytelnianie podstawowe; **NTLM** - używane jest uwierzytelnianie NTLM (Windows NT LAN Manager); **Kerberos** - używane jest uwierzytelnianie Kerberos (zobacz także: Konfigurowanie Kerberos z Zabbix); **Digest** - używane jest uwierzytelnianie Digest.
Użytkownik	Wprowadź nazwę użytkownika (do 255 znaków). To pole jest dostępne, jeśli dla opcji *Uwierzytelnianie HTTP* ustawiono Basic, NTLM, Kerberos lub Digest. Obsługiwane są makra użytkownika.
Hasło	Wprowadź hasło użytkownika (do 255 znaków). To pole jest dostępne, jeśli dla opcji *Uwierzytelnianie HTTP* ustawiono Basic, NTLM, Kerberos lub Digest. Obsługiwane są makra użytkownika.
Weryfikuj element równorzędny SSL	Zaznacz pole wyboru, aby zweryfikować certyfikat SSL serwera WWW. Certyfikat serwera jest weryfikowany względem certyfikatów zaufanych urzędów certyfikacji (CA) z systemowej lokalizacji CA. Lokalizację plików CA można zastąpić za pomocą parametru konfiguracji serwera Zabbix lub proxy SSLCALocation. Ustawia to opcję cURL CURLOPT_SSL_VERIFYPEER.
Weryfikuj host SSL	Zaznacz pole wyboru, aby zweryfikować, czy pole *Common Name* lub *Subject Alternate Name* certyfikatu serwera WWW jest zgodne. Ustawia to opcję cURL CURLOPT_SSL_VERIFYHOST.
Plik certyfikatu SSL	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta. Plik certyfikatu musi być w formacie PEM[1]. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw pole *Plik klucza SSL* puste. Jeśli klucz jest zaszyfrowany, określ hasło w polu *Hasło klucza SSL*. Katalog zawierający ten plik jest określany przez parametr konfiguracji serwera Zabbix lub proxy SSLCertLocation. W tym polu można używać makr `HOST.*` i makr użytkownika. Ustawia to opcję cURL CURLOPT_SSLCERT.
Plik klucza SSL	Nazwa pliku klucza prywatnego SSL używanego do uwierzytelniania klienta. Plik klucza prywatnego musi być w formacie PEM[1]. Katalog zawierający ten plik jest określany przez parametr konfiguracji serwera Zabbix lub proxy SSLKeyLocation. W tym polu można używać makr `HOST.*` i makr użytkownika. Ustawia to opcję cURL CURLOPT_SSLKEY.
Hasło klucza SSL	Hasło pliku klucza prywatnego SSL. W tym polu można używać makr użytkownika. Ustawia to opcję cURL CURLOPT_KEYPASSWD.

Attention:
[1] Zabbix obsługuje wyłącznie certyfikaty i pliki kluczy prywatnych w formacie PEM. Jeśli dane certyfikatu i klucza prywatnego znajdują się w pliku w formacie PKCS #12 (zwykle z rozszerzeniem *.p12 lub *.pfx), można wygenerować z niego plik PEM za pomocą następujących poleceń:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes  -out ssl-cert.key
```

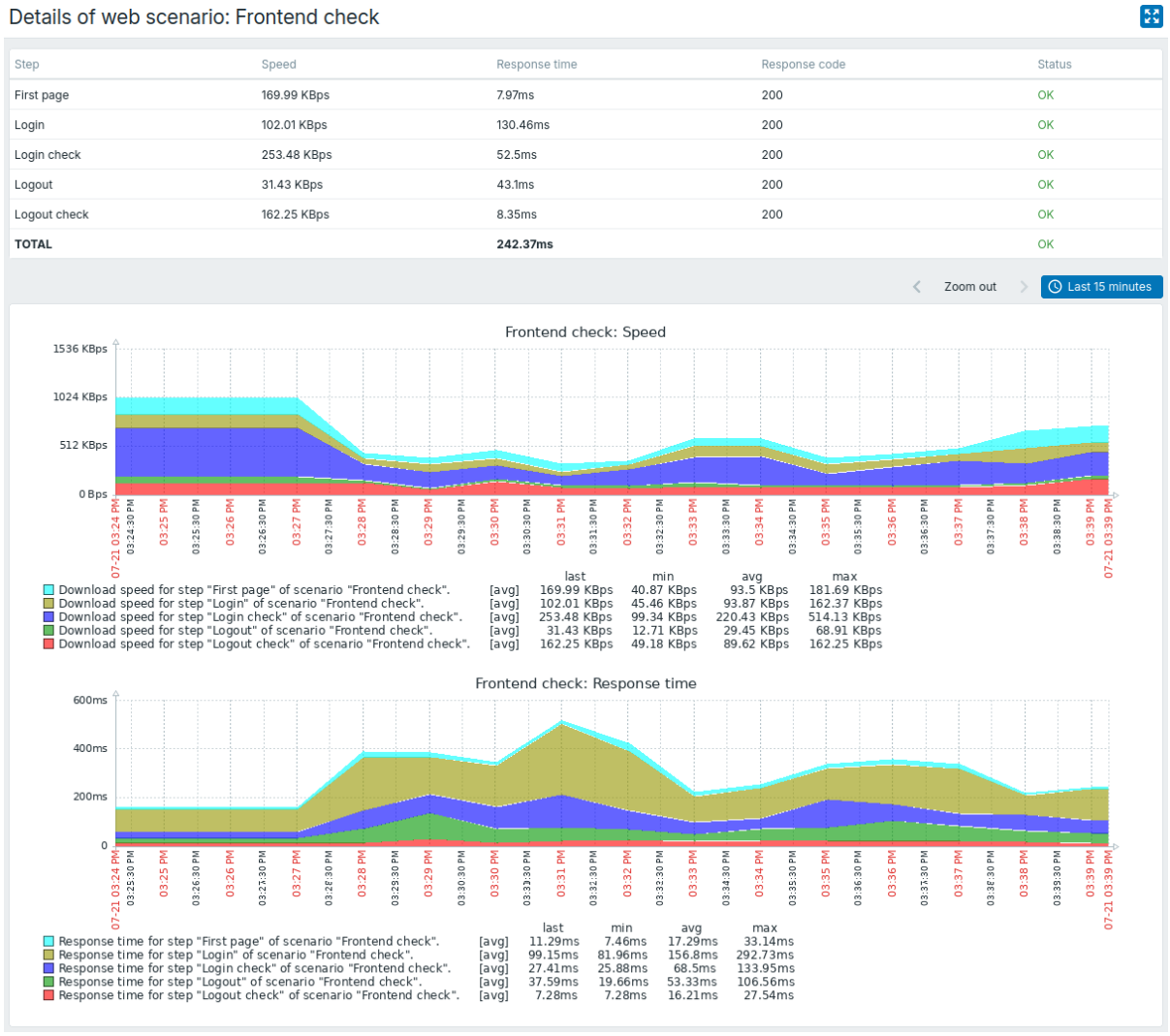
Note:
Serwer Zabbix wykrywa zmiany w certyfikatach bez ponownego uruchamiania.

Note:
Jeśli certyfikat klienta i klucz prywatny znajdują się w jednym pliku, określ go w polu „Plik certyfikatu SSL" i pozostaw pole „Plik klucza SSL" puste. Certyfikat i klucz nadal muszą być w formacie PEM. Połączenie certyfikatu i klucza jest proste:

```
cat client.crt client.key > client.pem
```

Wyświetlanie Aby wyświetlić scenariusze internetowe skonfigurowane dla host:

1. Przejdź do *Monitoring* > *Hosts*.
2. Znajdź host na liście i kliknij hiperłącze *Web* w ostatniej kolumnie.
3. Kliknij nazwę scenariusza, aby wyświetlić szczegółowe informacje.



Przegląd scenariuszy internetowych można również wyświetlić w sekcji *Dashboards* za pomocą widżetu monitorowania sieci Web.

Najnowsze wyniki wykonywania scenariusza internetowego są dostępne w sekcji *Monitoring* > *Latest data*.

Rozszerzone monitorowanie Czasami konieczne jest rejestrowanie otrzymanej zawartości strony HTML. Jest to szczególnie przydatne, jeśli któryś krok scenariusza webowego zakończy się niepowodzeniem. Do tego celu służy poziom debugowania 5 (trace). Ten poziom można ustawić w plikach konfiguracyjnych serwer i proxy lub za pomocą opcji runtime control (-R

`log_level_increase="http poller,N"`, gdzie N jest numerem procesu). Poniższe przykłady pokazują, jak można uruchomić rozszerzone monitorowanie, pod warunkiem że poziom debugowania 4 jest już ustawiony:

```
# 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"
```

Jeśli rozszerzone monitorowanie webowe nie jest wymagane, można je zatrzymać za pomocą opcji `-R log_level_decrease`.

Pozycje monitorowania sieci web

Przegląd

Niektóre nowe pozycje są automatycznie dodawane do monitorowania podczas tworzenia scenariuszy WWW.

Wszystkie pozycje dziedziczą tagi ze scenariusza WWW.

Pozycje scenariusza

Gdy tylko scenariusz zostanie utworzony, Zabbix automatycznie dodaje następujące pozycje do monitorowania.

Pozycja	Opis
Prędkość pobierania dla scenariusza <Scenario>	Ta pozycja będzie zbierać informacje o prędkości pobierania (bajty na sekundę) dla całego scenariusza, tj. średnią dla wszystkich kroków. Klucz pozycji: web.test.in[Scenario„bps] Typ: *Numeric(float)*
Nieudany krok scenariusza <Scenario>	Ta pozycja będzie wyświetlać numer kroku, który zakończył się niepowodzeniem w scenariuszu. Jeśli wszystkie kroki zostaną wykonane pomyślnie, zwracane jest 0. Klucz pozycji: web.test.fail[Scenario] Typ: *Numeric(unsigned)*
Treść ostatniego komunikatu o błędzie scenariusza <Scenario>	Ta pozycja zwraca treść ostatniego komunikatu o błędzie scenariusza. Nowa wartość jest zapisywana tylko wtedy, gdy scenariusz ma nieudany krok. Jeśli wszystkie kroki są poprawne, żadna nowa wartość nie jest zbierana. Klucz pozycji: web.test.error[Scenario] Typ: *Character*

Zamiast „Scenario” zostanie użyta rzeczywista nazwa scenariusza.

Note:
Jeśli nazwa scenariusza zawiera makra użytkownika, makra te pozostaną nierozwinięte w nazwach pozycji monitorowania WWW.

 Jeśli nazwa scenariusza zaczyna się od cudzysłowu lub zawiera przecinek albo nawias kwadratowy, zostanie poprawnie ujęta w cudzysłowy w kluczach pozycji. W pozostałych przypadkach nie zostanie zastosowane dodatkowe ujmowanie w cudzysłowy.

Note:
Pozycje monitorowania WWW są dodawane z 30-dniowym okresem przechowywania historii i 90-dniowym okresem przechowywania trendów.

Te pozycje mogą być używane do tworzenia wyzwalaczy i definiowania warunków powiadomień.

Przykład 1

Aby utworzyć wyzwalacz „Web scenario failed”, możesz zdefiniować wyrażenie wyzwalacza:

```
last(/host/web.test.fail[Scenario])<>0
```

Pamiętaj, aby zastąpić „Scenario” rzeczywistą nazwą swojego scenariusza.

Przykład 2

Aby utworzyć wyzwalacz „Web scenario failed” z użytecznym opisem problemu w nazwie wyzwalacza, możesz zdefiniować wyzwalacz o nazwie:

`Web scenario "Scenario" failed: {ITEM.VALUE}`

oraz wyrażeniu wyzwalacza:

`length(last(/host/web.test.error[Scenario]))>0 and last(/host/web.test.fail[Scenario])>0`

Pamiętaj, aby zastąpić „Scenario” rzeczywistą nazwą scenariusza.

Przykład 3

Aby utworzyć wyzwalacz „Aplikacja web jest wolna”, możesz zdefiniować wyrażenie wyzwalacza:

`last(/host/web.test.in[Scenario,,bps])<10000`

Pamiętaj, aby zastąpić „Scenario” rzeczywistą nazwą swojego scenariusza.

Pozycje kroków scenariusza

Gdy tylko krok zostanie utworzony, Zabbix automatycznie dodaje następujące pozycje do monitorowania.

Pozycja	Opis
Prędkość pobierania dla kroku <Step> scenariusza <Scenario>	Ta pozycja będzie zbierać informacje o prędkości pobierania (bajty na sekundę) dla kroku. Klucz pozycji: web.test.in[Scenario,Step,bps] Typ: *Numeric(float)*
Czas odpowiedzi dla kroku <Step> scenariusza <Scenario>	Ta pozycja będzie zbierać informacje o czasie odpowiedzi kroku w sekundach. Czas odpowiedzi jest liczony od początku żądania do momentu przesłania wszystkich informacji. Klucz pozycji: web.test.time[Scenario,Step,resp] Typ: *Numeric(float)*
Kod odpowiedzi dla kroku <Step> scenariusza <Scenario>	Ta pozycja będzie zbierać kody odpowiedzi kroku. Klucz pozycji: web.test.rspcode[Scenario,Step] Typ: *Numeric(unsigned)*

Zamiast ”Scenario” i ”Step” zostaną użyte odpowiednio rzeczywiste nazwy scenariusza i kroku.

Note:
Pozycje monitorowania WWW są dodawane z 30-dniowym okresem przechowywania historii i 90-dniowym okresem przechowywania trendów.

Note:
Jeśli nazwa scenariusza zaczyna się od cudzysłowu lub zawiera przecinek albo nawias kwadratowy, zostanie poprawnie ujęta w cudzysłowy w kluczach pozycji. W innych przypadkach nie zostanie zastosowane dodatkowe ujmowanie w cudzysłowy.

Te pozycje mogą być używane do tworzenia wyzwalaczy i definiowania warunków powiadomień. Na przykład, aby utworzyć wyzwalacz ”Logowanie do GUI Zabbix jest zbyt wolne”, można zdefiniować wyrażenie wyzwalacza:

`last(/zabbix/web.test.time[ZABBIX GUI,Login,resp])>3`

Scenariusz z życia wzięty

Przegląd

Ta sekcja przedstawia krok po kroku rzeczywisty przykład użycia monitorowania web.

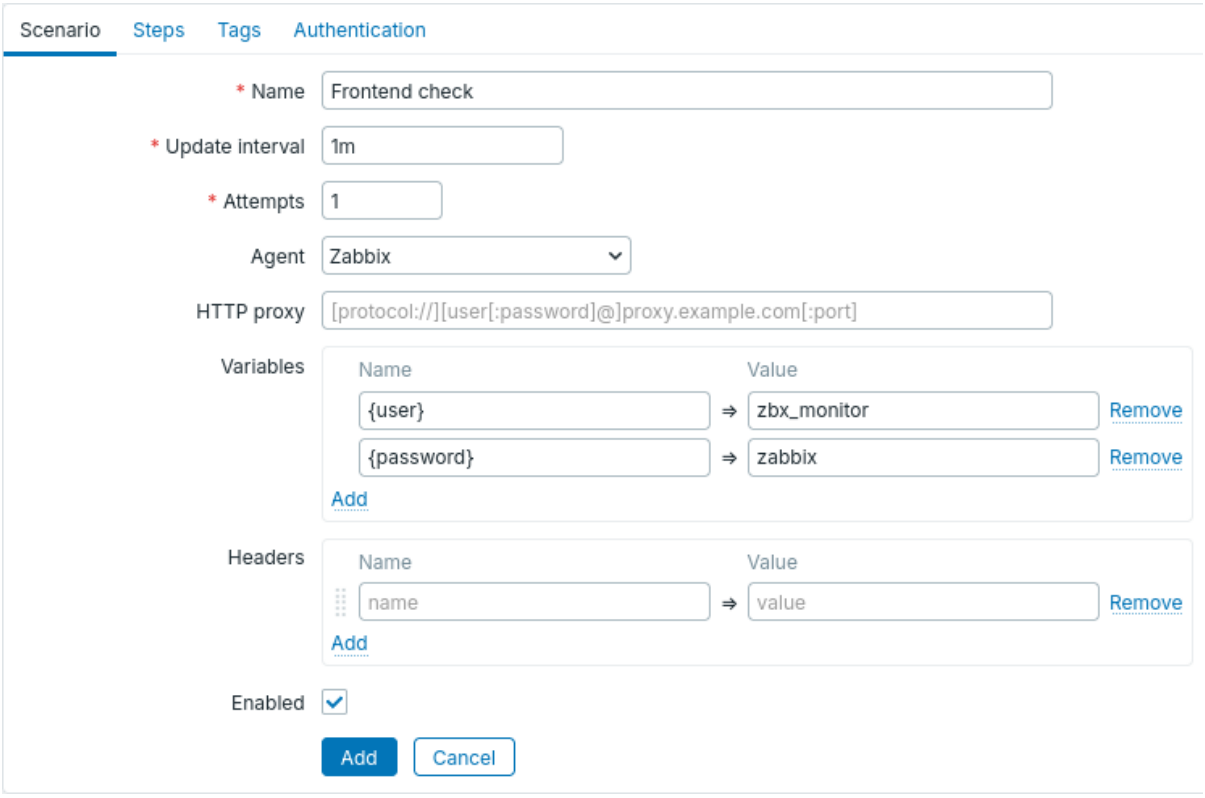
Do monitorowania frontend Zabbix zostanie użyte monitorowanie web Zabbix. Celem jest określenie, czy jest on dostępny, czy udostępnia właściwą zawartość oraz jak szybko działa. Aby to osiągnąć, wymaganych jest kilka kroków, w tym sprawdzenie dostępności pierwszej strony, logowanie przy użyciu nazwy użytkownika i hasła, weryfikacja powodzenia logowania, wylogowanie oraz potwierdzenie wylogowania.

Scenariusz

Dodawanie nowego scenariusza web

1. Przejdź do *Data collection* > *Hosts*.
2. Wybierz host i kliknij *Web* w wierszu tego hosta.

3. Kliknij *Create web scenario*.



4. W nowym formularzu scenariusza, na karcie *Scenario*, wypełnij następujące pola:

    - *Name*: Frontend check.
    - *Update interval*: 1m.
    - *Attempts*: 1.
    - *Agent*: Zabbix.
 W sekcji *Variables* dodaj następujące pary nazwa-wartość:
    - `{user}`: Twoje istniejące dane uwierzytelniające użytkownika Zabbix;
    - `{password}`: Twoje istniejące dane uwierzytelniające użytkownika Zabbix.

Note:
Ze względów bezpieczeństwa zaleca się utworzenie osobnego użytkownika z minimalnymi uprawnieniami do wykorzystania w celach monitorowania.

Opcjonalnie przejdź do karty *Tags* i dodaj tagi scenariusza web. Tagów scenariusza web można używać do szybkiego identyfikowania powiązanych pozycji i wyzwalaczy lub przeszukiwania zebranych danych. Na przykład odpowiednie tagi dla tego samouczka to `component: web-scenario` i/lub `target: frontend`.

Konfigurowanie kroków scenariusza webowego

Przejdź do karty *Steps* i zdefiniuj kroki dla scenariusza.
 Po skonfigurowaniu każdego kroku kliknij przycisk *Add*.

Attention:
Należy pamiętać, że frontend Zabbixa używa przekierowania JavaScript podczas logowania, więc logowanie musi nastąpić jako pierwsze, a funkcje dostępne po zalogowaniu można sprawdzać dopiero w kolejnych krokach. Dodatkowo krok logowania musi używać pełnego adresu URL do pliku **index.php**.

Wspólne pola dla wszystkich kroków

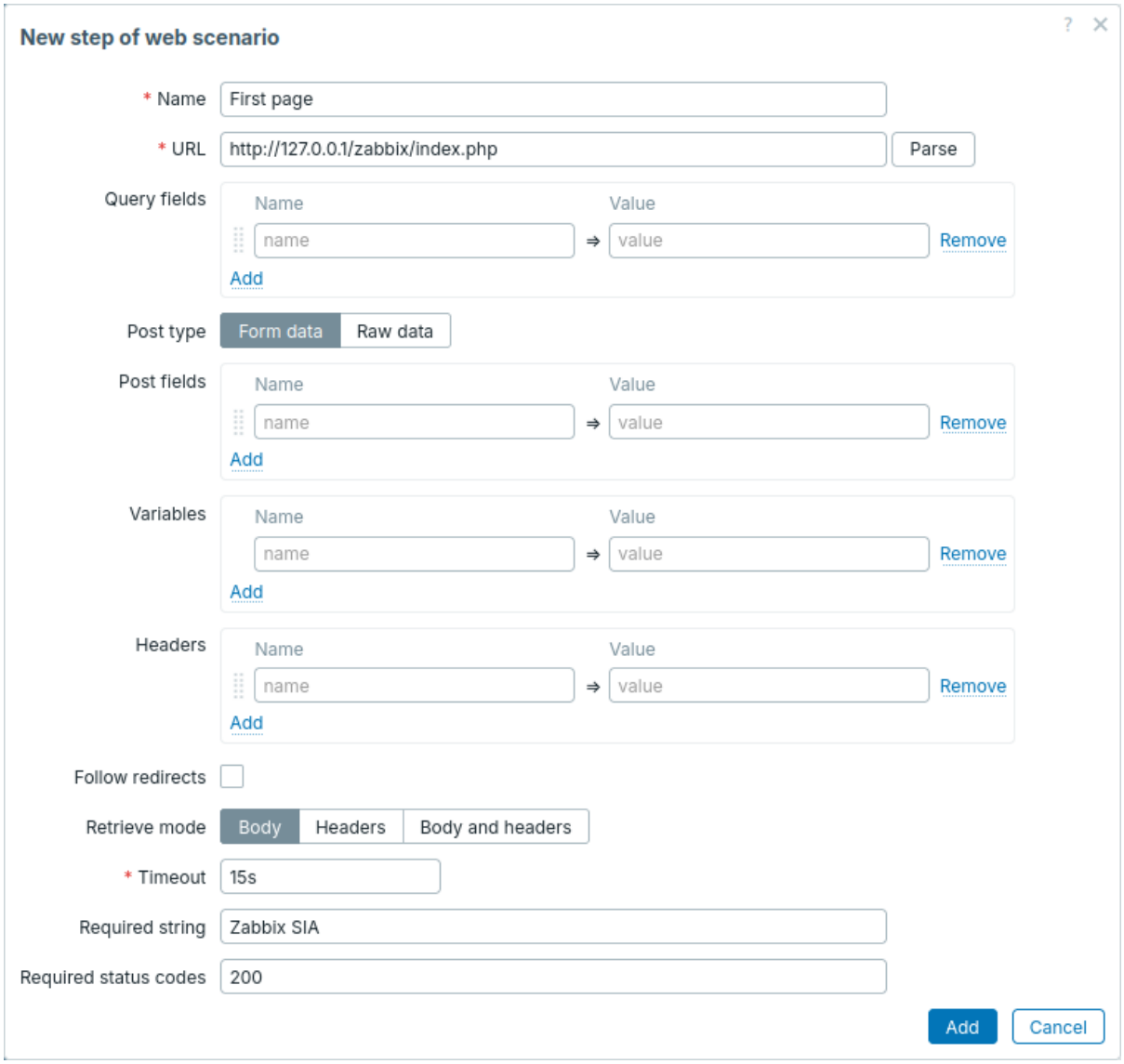
Oprócz pól specyficznych dla danego kroku wypełnij dla każdego z opisanych poniżej kroków następujące wspólne pola:

- *URL*: adres URL frontendu Zabbix (`http://<server_ip_or_name>/zabbix/index.php` (dla Apache) lub `http://<server_ip_or` (dla NGINX)). W kroku 3 scenariusza internetowego użyj innego adresu URL, zgodnie z poniższym opisem.
- *Timeout*: 15s.
- *Required status codes*: 200.

Krok 1 scenariusza web

Sprawdź, czy pierwsza strona odpowiada poprawnie, zwraca kod statusu HTTP 200 i zawiera tekst *Zabbix SIA*.
 Wypełnij następujące pola:

- *Name*: First page.
- *Timeout*: 15s.
- *Required string*: Zabbix SIA.
- Wspólne pola dla wszystkich kroków.

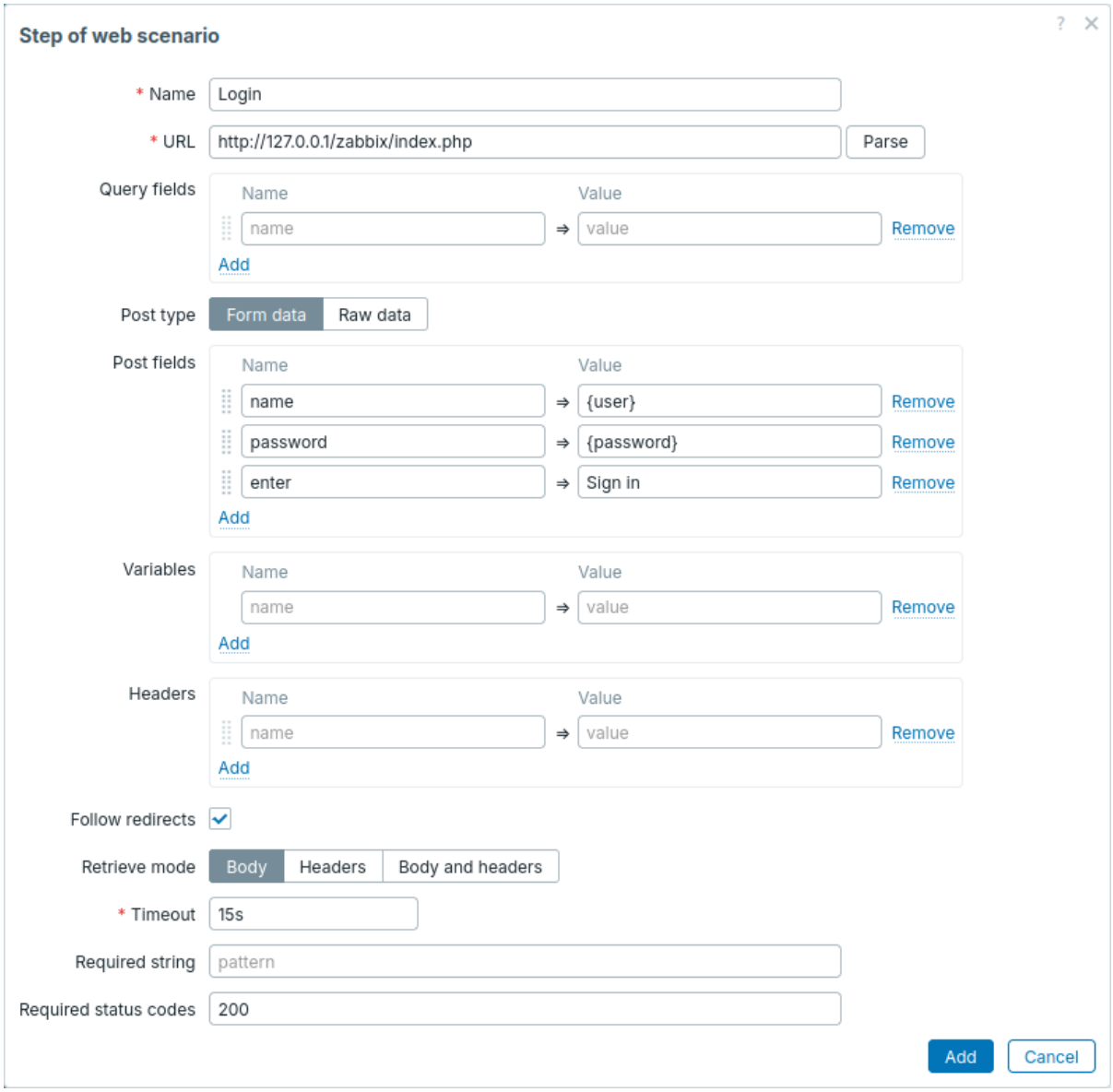


Krok 2 scenariusza WWW

Zaloguj się do frontend Zabbix, używając makr (zmiennych) zdefiniowanych na karcie *Scenario*: *{user}* i *{password}*.
 Wypełnij następujące pola:

- *Name*: Login.
- W sekcji *Post fields* dodaj następujące pary nazwa-wartość:
  - name: `{user}`;
  - password: `{password}`;
  - enter: Sign in.
- Wspólne pola dla wszystkich kroków.

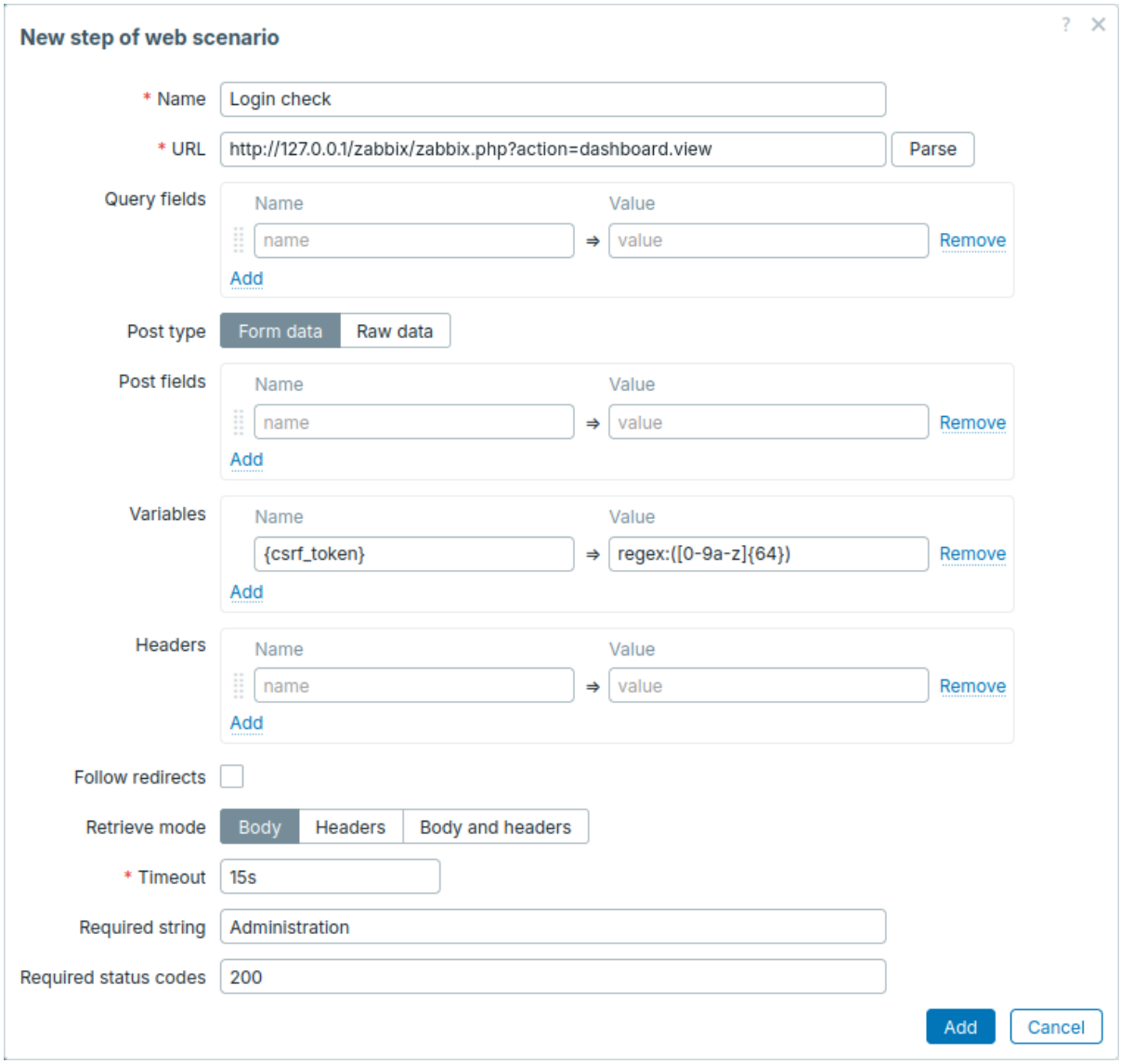
Zaznacz pole wyboru *Follow redirects*.



Krok 3 scenariusza web

Po zalogowaniu sprawdź powodzenie, wyszukując ciąg widoczny tylko po zalogowaniu, na przykład *Administration*.
 Wypełnij następujące pola:

- *Name*: Login check.
- *URL*: adres URL, na który użytkownik jest przekierowywany po zalogowaniu (na przykład http://127.0.0.1/zabbix/zabbix.php?action=dash
- W sekcji *Variables* dodaj parę nazwa-wartość `{csrf_token}`: `regex:([0-9a-z]{64})`. Ta zmienna przechwyci wartość przypisanego tokenu CSRF, aby można było go ponownie użyć w kroku 4 scenariusza web.
- *Required string*: Administration.
- Wspólne pola dla wszystkich kroków.



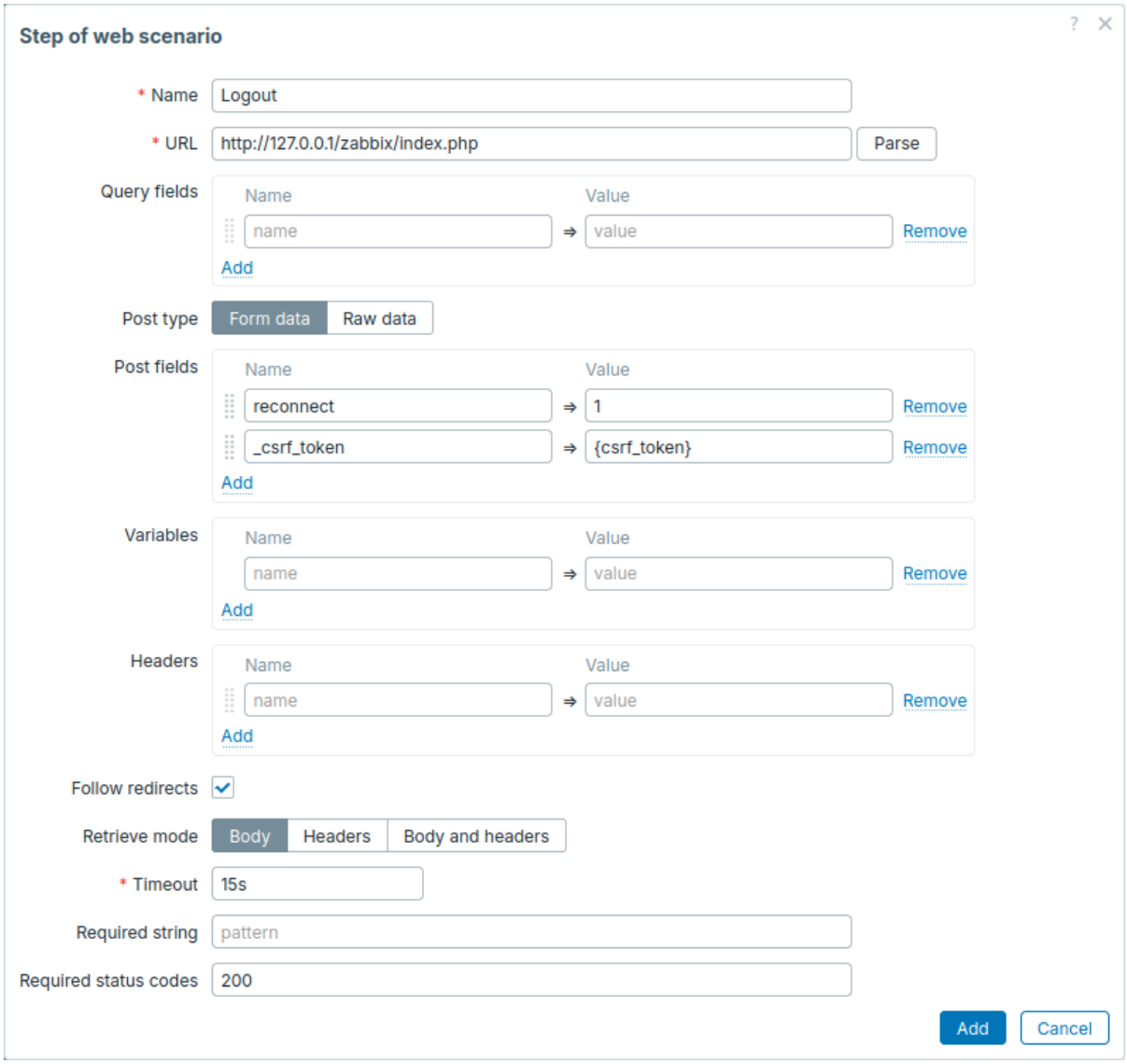
Krok 4 scenariusza webowego

Po zweryfikowaniu dostępności frontend i logowania dodaj krok wylogowania; w przeciwnym razie baza danych Zabbix zostanie zaśmiecona wieloma otwartymi rekordami sesji.
 Wypełnij następujące pola:

- *Nazwa*: Wylogowanie.
- W sekcji *Pola POST* dodaj następujące pary nazwa-wartość:
  - reconnect: 1;
  - _csrf_token: `{csrf_token}`.
- Wspólne pola dla wszystkich kroków.

Zaznacz pole wyboru *Podążaj za przekierowaniami*.

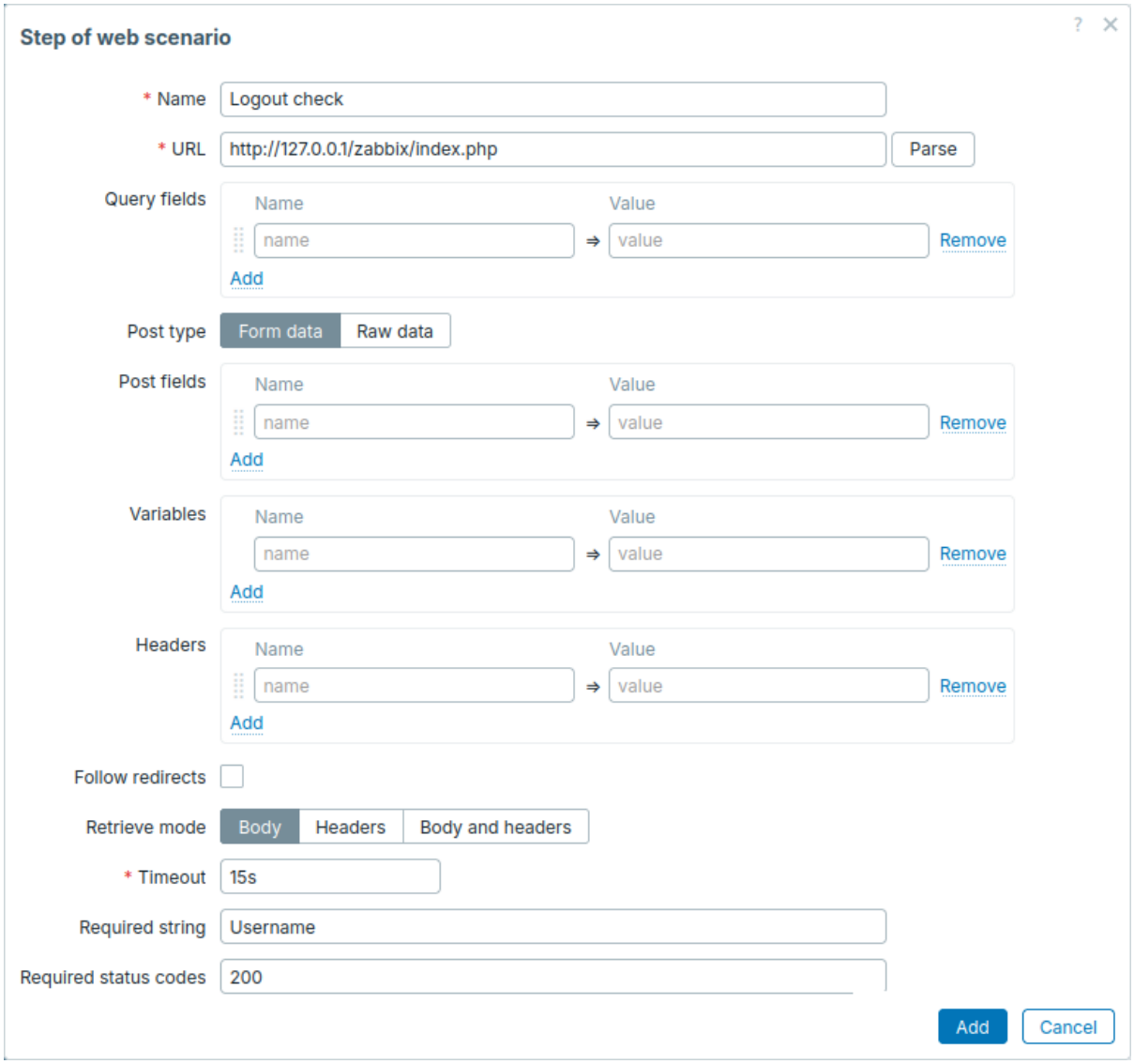
Ten krok używa zmiennej `{csrf_token}` uzyskanej w kroku 3 scenariusza webowego.



Krok 5 scenariusza web

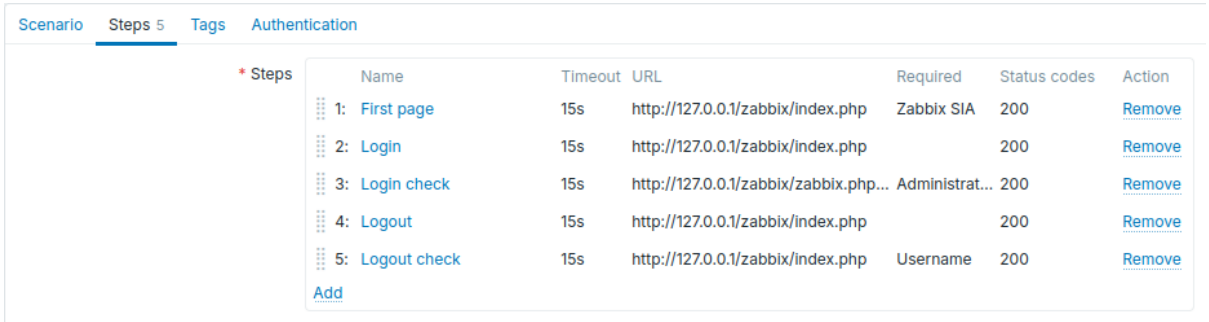
Aby potwierdzić wylogowanie, sprawdź ciąg *Username*.
 Wypełnij następujące pola:

- *Name*: Logout check.
- *Required string*: Username.
- Wspólne pola dla wszystkich kroków.



Pełna konfiguracja kroków

Pełna konfiguracja kroków scenariusza WWW powinna wyglądać następująco:

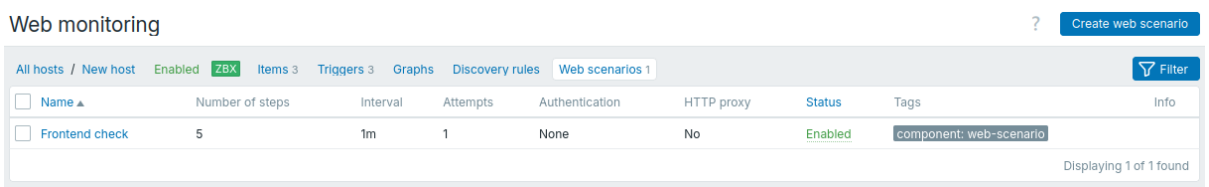


Sprawdź wyniki

Zapisz ukończony scenariusz monitorowania WWW.

Gdy scenariusz WWW zostanie utworzony, nowe pozycje monitorowania WWW są automatycznie dodawane do monitorowania.

Scenariusz zostanie dodany do host. Aby wyświetlić informacje o scenariuszu WWW, przejdź do *Monitoring* > *Hosts*, znajdź host na liście i kliknij hiperłącze *Web* w ostatniej kolumnie.



Kliknij nazwę scenariusza, aby zobaczyć szczegółowe statystyki:



Monitorowanie usług

Przegląd Monitorowanie usług to monitorowanie na poziomie biznesowym, które można wykorzystać do uzyskania przeglądu całego drzewa usług infrastruktury IT, identyfikowania słabych punktów infrastruktury, obliczania SLA różnych usług IT oraz sprawdzania innych informacji na wyższym poziomie. Monitorowanie usług koncentruje się na ogólnej dostępności usługi zamiast na szczegółach niskiego poziomu, takich jak brak miejsca na dysku, wysokie obciążenie procesora itp. Monitorowanie usług zapewnia również funkcję znajdowania głównej przyczyny problemu, jeśli usługa nie działa zgodnie z oczekiwaniami.

Monitorowanie usług umożliwia tworzenie hierarchicznej reprezentacji monitorowanych danych.

Bardzo prosta struktura usług może wyglądać następująco:

```
Usługa
|
|-Stacje robocze
| |
| |-Stacja robocza1
| |
| |-Stacja robocza2
|
|-Serwery
```

Każdy węzeł struktury ma atrybut stanu. Stan jest obliczany i propagowany na wyższe poziomy zgodnie z wybranym algorytmem. Na stan poszczególnych węzłów wpływa stan zmapowanych problemów. Mapowanie problemów odbywa się za pomocą tagowania.

Zabbix może wysyłać powiadomienia lub automatycznie wykonywać skrypt na serwerze Zabbix w przypadku wykrycia zmiany stanu usługi. Możliwe jest zdefiniowanie elastycznych reguł określających, czy usługa nadrzędna powinna przejść w „Stan problemu”

na podstawie stanów usług podrzędnych. Dane dotyczące problemów usług można następnie wykorzystać do obliczania SLA i wysyłania raportów SLA na podstawie elastycznego zestawu warunków.

Monitorowanie usług konfiguruje się w menu Usługi, które składa się z następujących sekcji:

- Usługi

Sekcja Usługi umożliwia zbudowanie hierarchii monitorowanej infrastruktury poprzez dodawanie usług nadrzędnych, a następnie usług podrzędnych do usług nadrzędnych.

Oprócz konfigurowania drzewa usług sekcja ta zapewnia przegląd całej infrastruktury i umożliwia szybkie identyfikowanie problemów, które doprowadziły do zmiany stanu usługi.

- SLA

W tej sekcji można definiować umowy dotyczące poziomu usług i ustawiać cele poziomu usług dla określonych usług.

- Raport SLA

W tej sekcji można wyświetlać raporty SLA.

Działania usług

Można również skonfigurować działania usług.

Działania usług są opcjonalne i umożliwiają:

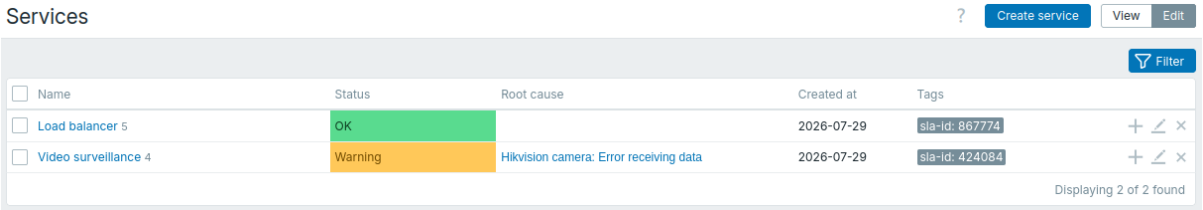
- wysyłanie powiadomienia o niedostępności usługi
- wykonywanie zdalnego polecenia na serwerze Zabbix po zmianie stanu usługi
- wysyłanie powiadomienia o przywróceniu działania usługi.

Zobacz także:

- Przykład konfiguracji monitorowania SLA
- Uwagi dotyczące aktualizacji usług z wersji Zabbix starszych niż 6.0

Drzewo usług

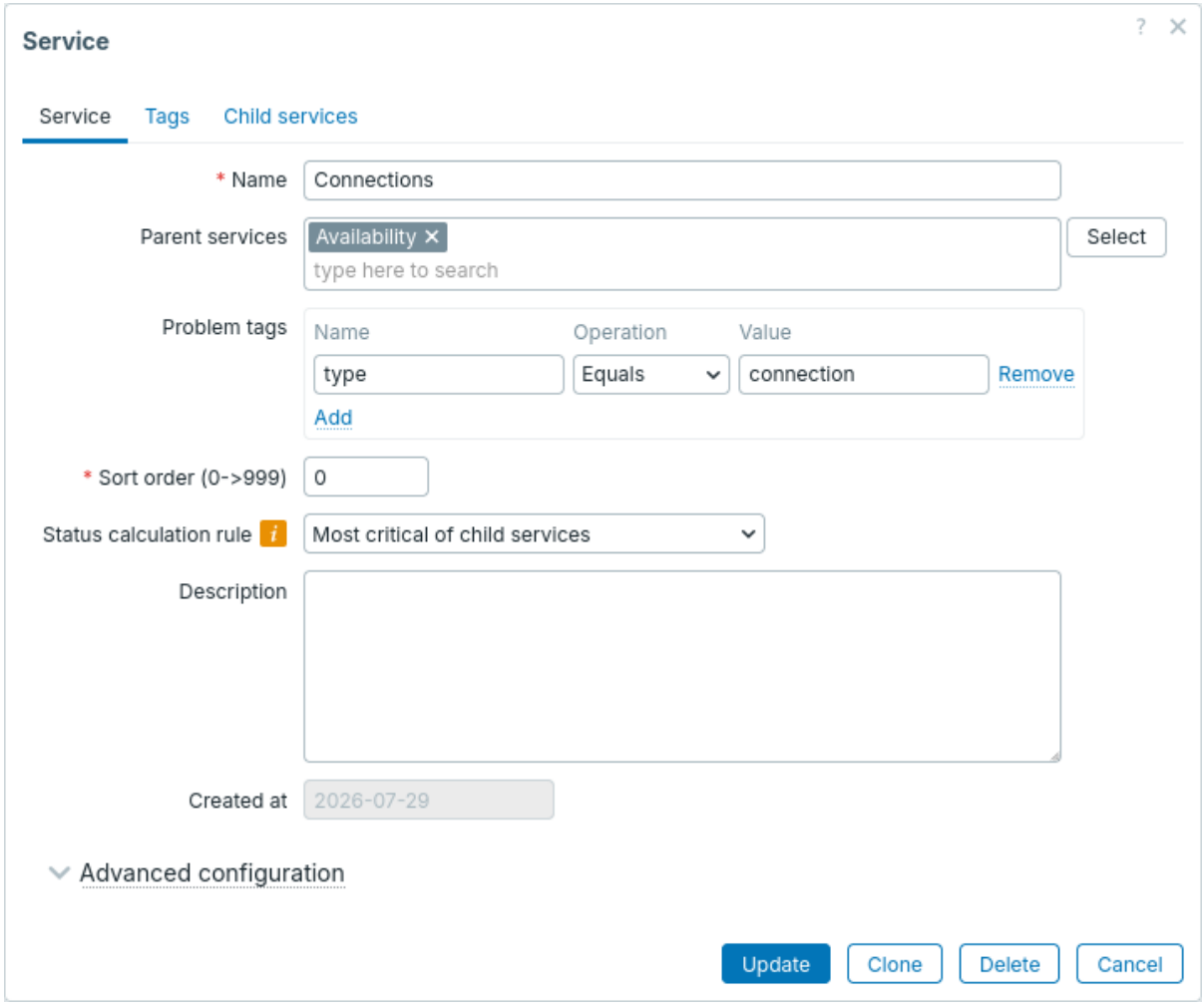
Drzewo usług jest konfigurowane w sekcji menu *Services* > *Services*. W prawym górnym rogu przełącz z trybu view do trybu edycji.



Aby skonfigurować nową usługę, kliknij przycisk *Create service* w prawym górnym rogu.

Aby szybko dodać usługę podrzędną, możesz też kliknąć ikonę plusa obok usługi nadrzędnej. Spowoduje to otwarcie tego samego formularza konfiguracji usługi, ale parametr *Parent services* będzie wstępnie wypełniony.

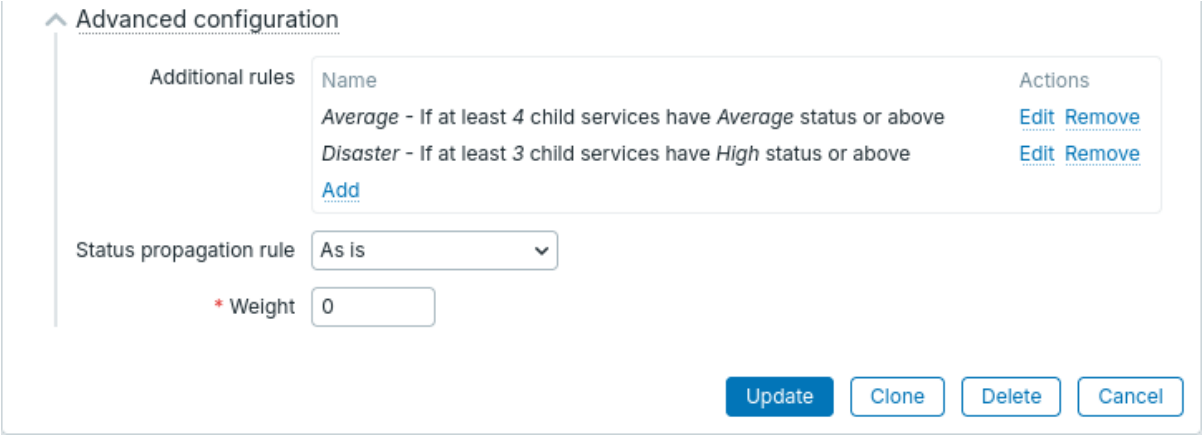
Konfiguracja usługi Na karcie **Usługa** określ wymagane parametry usługi:



Wszystkie obowiązkowe pola wejściowe są oznaczone czerwoną gwiazdką.

Parameter	Description
Name	Nazwa usługi.
Parent services	Usługi nadrzędne, do których należy usługa. Pozostaw to pole puste, jeśli dodajesz usługę najwyższego poziomu. Jedna usługa może mieć wiele usług nadrzędnych. W takim przypadku będzie wyświetlana w drzewie usług pod każdą z usług nadrzędnych.
Problem tags	Określ tagi, aby mapować dane problemów na usługę. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są dwa operatory: **Equals** - uwzględnij określone nazwy i wartości tagów (rozróżnianie wielkości liter) **Contains** - uwzględnij określone nazwy tagów, w których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, rozróżnianie wielkości liter)
Sort order	Kolejność sortowania wyświetlania, najniższa wartość jest pierwsza.
Status calculation rule	Reguła obliczania statusu usługi: **Most critical if all children have problems** - najbardziej krytyczny problem w usługach podrzędnych jest używany do oznaczenia kolorem statusu usługi, jeśli wszystkie usługi podrzędne mają problemy **Most critical of child services** - najbardziej krytyczny problem w usługach podrzędnych jest używany do oznaczenia kolorem statusu usługi **Set status to OK** - nie obliczaj statusu usługi Dodatkowe reguły obliczania statusu można skonfigurować w opcjach advanced configuration.
Description	Opis usługi.
Created at	Czas utworzenia usługi; wyświetlany podczas edycji istniejącej usługi.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje advanced configuration.

Zaawansowana konfiguracja



Parameter	Opis
Additional rules	Kliknij *Add*, aby skonfigurować dodatkowe reguły obliczania statusu.
Set status to	Ustaw status usługi na *OK* (domyślnie), *Not classified*, *Information*, *Warning*, *Average*, *High* lub *Disaster* w przypadku spełnienia warunku.
Condition	Wybierz warunek dla bezpośrednich usług podrzędnych: **jeśli co najmniej (N) usług podrzędnych ma status (Status) lub wyższy** **jeśli co najmniej (N%) usług podrzędnych ma status (Status) lub wyższy** **jeśli mniej niż (N) usług podrzędnych ma status (Status) lub niższy** **jeśli mniej niż (N%) usług podrzędnych ma status (Status) lub niższy** **jeśli waga usług podrzędnych ze statusem (Status) lub wyższym wynosi co najmniej (W)** **jeśli waga usług podrzędnych ze statusem (Status) lub wyższym wynosi co najmniej (N%)** **jeśli waga usług podrzędnych ze statusem (Status) lub niższym jest mniejsza niż (W)** **jeśli waga usług podrzędnych ze statusem (Status) lub niższym jest mniejsza niż (N%)** Jeśli określono kilka warunków i sytuacja spełnia więcej niż jeden z nich, zostanie ustawiony najwyższy poziom ważności.
N (W)	Ustaw wartość N lub W (1-100000) albo N% (1-100) w warunku.
Status	Wybierz wartość *Status* w warunku: *OK* (domyślnie), *Not classified*, *Information*, *Warning*, *Average*, *High* lub *Disaster*.
Status propagation rule	Reguła propagowania statusu usługi do usługi nadrzędnej: **As is** - status jest propagowany bez zmian **Increase by** - można zwiększyć propagowany status o 1 do 5 poziomów ważności **Decrease by** - można zmniejszyć propagowany status o 1 do 5 poziomów ważności **Ignore this service** - status nie jest w ogóle propagowany do usługi nadrzędnej **Fixed status** - status jest propagowany statycznie, tj. zawsze taki sam
Weight	Waga usługi (liczba całkowita w zakresie od 0 (domyślnie) do 1000000).

Note:
Dodatkowe reguły obliczania statusu mogą być używane wyłącznie do zwiększania poziomu ważności względem poziomu obliczonego zgodnie z głównym parametrem *Status calculation rule*. Jeśli zgodnie z dodatkowymi regułami status powinien być Warning, ale zgodnie z parametrem *Status calculation rule* status to Disaster - usługa będzie miała status Disaster.

Karta *Tags* zawiera tagi na poziomie usługi. Tagi na poziomie usługi służą do identyfikowania usługi. Tagi tego typu nie są używane do mapowania problemów na usługę (w tym celu użyj opcji *Tagi problemów* z pierwszej karty).

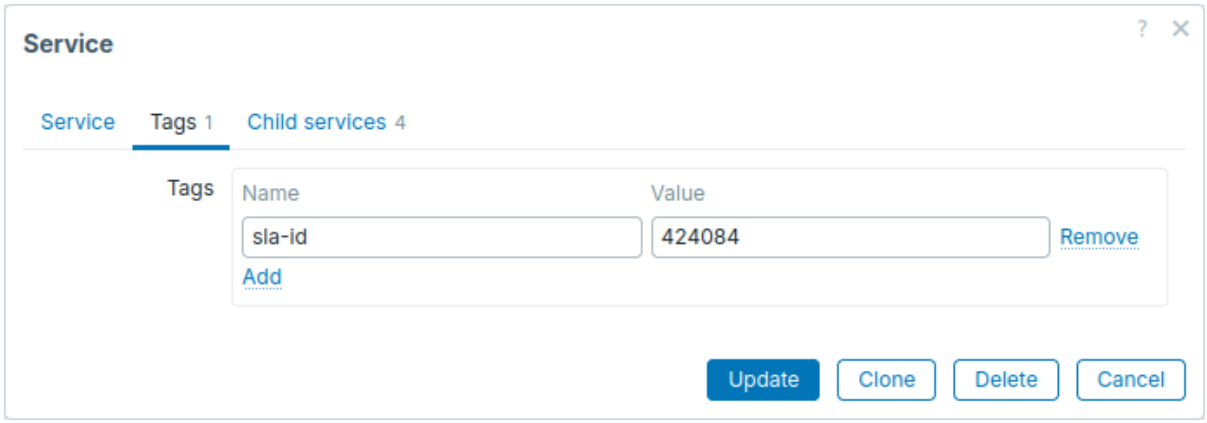
Karta *Usługi podrzędne* umożliwia określenie usług zależnych. Kliknij *Dodaj*, aby dodać usługę z listy istniejących usług. Jeśli chcesz dodać nową usługę podrzędną, najpierw zapisz tę usługę, a następnie kliknij ikonę plusa obok właśnie utworzonej usługi.

Tagi W usługach istnieją dwa różne typy tagów:

- Tagi usługi
- Tagi problemu

Tagi usługi

Tagi usługi są używane do dopasowywania usług do akcji usługi oraz SLA. Tagi te są określane na karcie konfiguracji usługi *Tagi*. Do mapowania SLA używana jest logika *OR*: usługa zostanie przypisana do SLA, jeśli ma co najmniej jeden pasujący tag. W akcjach usługi reguły mapowania są konfigurowalne i mogą używać logiki *AND*, *OR* lub *AND/OR*.

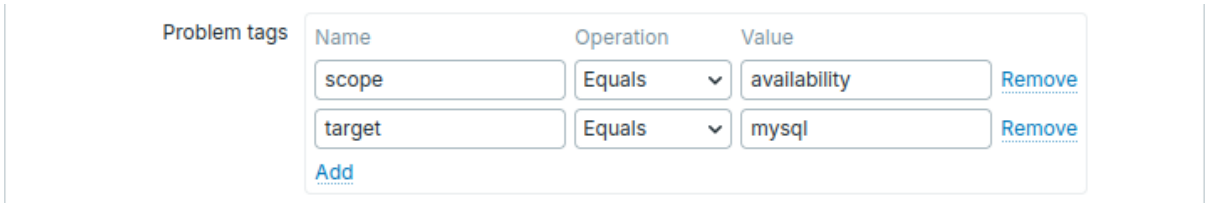


Tagi problemów

Tagi problemów są używane do dopasowywania problemów i usług. Tagi te są określane na głównej karcie konfiguracji usługi.

Tylko usługi podrzędne najniższego poziomu hierarchii mogą mieć zdefiniowane tagi problemów i być bezpośrednio powiązane z problemami. Jeśli tagi problemów pasują, status usługi zmieni się na taki sam, jaki ma problem. W przypadku kilku problemów usługa będzie miała status najbardziej krytycznego z nich. Status usługi nadrzędnej jest następnie obliczany na podstawie statusów usług podrzędnych zgodnie z regułami obliczania statusu.

Jeśli określono kilka tagów, używana jest logika *AND*: problem musi mieć wszystkie tagi określone w konfiguracji usługi, aby został przypisany do usługi.

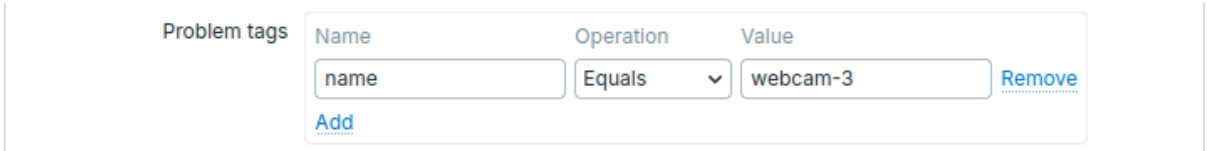


Note:
Problem w Zabbix dziedziczy tagi z całego łańcucha szablonów, hostów, pozycji, scenariuszy web i wyzwalaczy. Do dopasowywania problemów do usług można użyć dowolnego z tych tagów.

Przykład:

Problem *Web camera 3 is down* ma tagi `type:video-surveillance`, `floor:1` i `name:webcam-3` oraz status *Warning*

Usługa **Web camera 3** ma określony tylko jeden tag problemu: `name:webcam-3`



Status usługi zmieni się z *OK* na *Warning*, gdy ten problem zostanie wykryty.

Jeśli usługa **Web camera 3** miałaby tagi problemu `name:webcam-3` i `floor:2`, jej status nie zostałby zmieniony po wykryciu problemu, ponieważ warunki są spełnione tylko częściowo.

Modyfikowanie istniejących usług

Note:
Przyciski opisane poniżej są widoczne tylko wtedy, gdy sekcja *Services* jest w trybie edycji.

Aby edytować istniejącą usługę, kliknij ikonę ołówka obok usługi.

Aby sklonować istniejącą usługę, kliknij ikonę ołówka, aby otworzyć jej konfigurację, a następnie kliknij przycisk *Clone*. Po sklonowaniu usługi jej linki nadrzędne są zachowywane, natomiast linki podrzędne nie.

Aby usunąć usługę, kliknij ikonę `x` obok niej. Po usunięciu usługi nadrzędnej jej usługi podrzędne nie zostaną usunięte i zostaną przeniesione o jeden poziom wyżej w drzewie usług (dzieci 1. poziomu otrzymają ten sam poziom co usunięta usługa nadrzędna).

Dwa przyciski poniżej listy usług oferują kilka opcji masowej edycji:

- *Mass update* - masowa aktualizacja właściwości usługi
- *Delete* - usuń usługi

Aby skorzystać z tych opcji, zaznacz pola wyboru przed odpowiednimi usługami, a następnie kliknij wymagany przycisk.

SLA

Przegląd Po utworzeniu usług możesz rozpocząć monitorowanie, czy ich wydajność jest zgodna z umową dotyczącą poziomu usług (SLA).

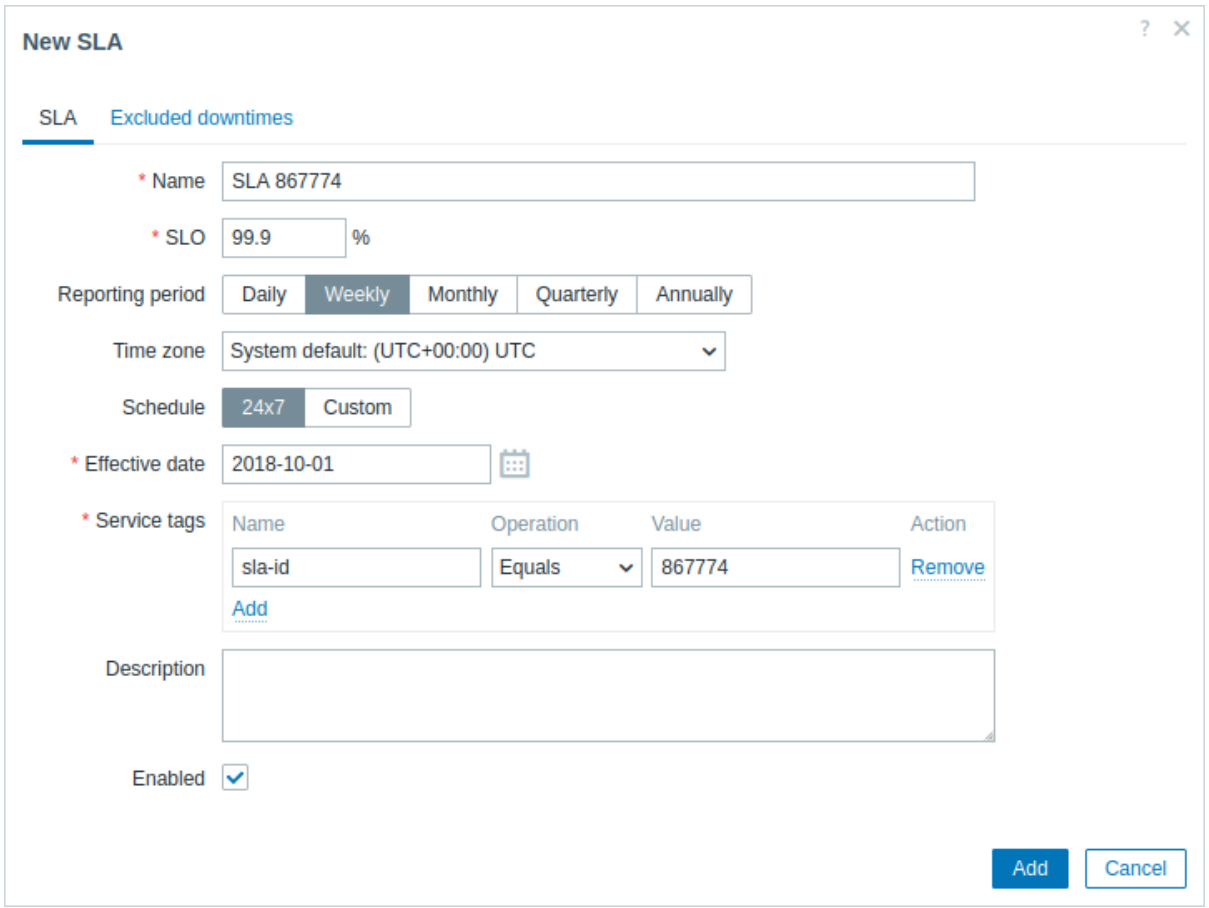
Sekcja menu *Usługi* > *SLA* umożliwia skonfigurowanie umów SLA dla różnych usług. SLA w Zabbix definiuje cel poziomu usług (SLO), oczekiwany harmonogram dostępności oraz planowane przestoje.

Umowy SLA i usługi są dopasowywane na podstawie znaczników usług. Ta sama umowa SLA może być stosowana do wielu usług — wydajność będzie mierzona osobno dla każdej pasującej usługi. Do pojedynczej usługi można przypisać wiele umów SLA — dane dla każdej z nich będą wyświetlane osobno.

W raportach SLA Zabbix udostępnia dane dotyczące wskaźnika poziomu usług (SLI), który mierzy rzeczywistą dostępność usługi. To, czy usługa spełnia cele SLA, jest określane przez porównanie SLO (oczekiwanej dostępności w %) z SLI (rzeczywistą dostępnością w %).

Konfiguracja Aby utworzyć nowy SLA, kliknij przycisk *Utwórz SLA*.

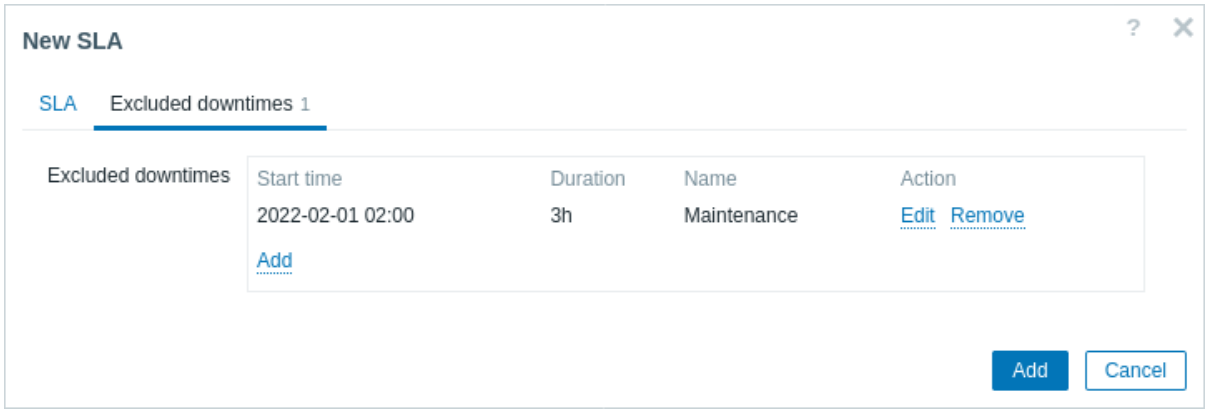
Karta *SLA* umożliwia określenie ogólnych parametrów SLA.



Parametr	Opis
Nazwa	Wprowadź nazwę SLA.
SLO	Wprowadź cel poziomu usług (SLO) w procentach.

Parametr	Opis
Okres raportowania	Wybierz okres raportowania SLI używany do obliczania danych SLI w raporcie SLA: **Dziennie** - każdy dzień, od 00:00:00 do 23:59:59. **Tygodniowo** - każdy tydzień, od niedzieli 00:00:00 do soboty 23:59:59. **Miesięcznie** - każdy miesiąc, od pierwszego dnia 00:00:00 do ostatniego dnia 23:59:59. **Kwartalnie** - każdy kwartał kalendarzowy: Q1 (sty-mar), Q2 (kwi-cze), Q3 (lip-wrz), Q4 (paź-gru). **Rocznie** - każdy rok kalendarzowy, od 1 stycznia 00:00:00 do 31 grudnia 23:59:59.
Strefa czasowa	Wybierz strefę czasową SLA.
Harmonogram	Wybierz harmonogram SLA - 24x7 lub niestandardowy.
Data rozpoczęcia obowiązywania	Wybierz datę rozpoczęcia obliczania SLA.
Tagi usługi	Dodaj tagi usługi, aby zidentyfikować usługi, do których powinno mieć zastosowanie to SLA. **Nazwa** - nazwa tagu usługi; musi być dokładnie zgodna, z uwzględnieniem wielkości liter. **Operacja** - wybierz *Równa się*, jeśli wartość tagu musi być dokładnie zgodna (z uwzględnieniem wielkości liter), lub *Zawiera*, jeśli zgodna musi być część wartości tagu (bez uwzględniania wielkości liter). **Wartość** - wartość tagu usługi, której należy szukać zgodnie z wybraną operacją. SLA jest stosowane do usługi, jeśli pasuje co najmniej jeden tag usługi.
Opis	Dodaj opis SLA.
Włączone	Zaznacz pole wyboru, aby włączyć obliczanie SLA.

Karta *Wykluczone przestoje* umożliwia określenie przestojów wykluczonych z obliczania SLA.



Kliknij *Dodaj*, aby skonfigurować wykluczone przestoje, a następnie wprowadź nazwę okresu, datę rozpoczęcia i czas trwania.

Raporty SLA Informacje o tym, jak usługa realizuje SLA, są widoczne w raporcie SLA. Raporty SLA można wyświetlić:

- w sekcji *SLA*, klikając hiperłącze raportu SLA;
- w sekcji *Services*, klikając nazwę SLA na karcie informacji;
- w widżecie *SLA report* na pulpicie.

Po skonfigurowaniu SLA karta *Info* w sekcji usług będzie również wyświetlać pewne informacje o wydajności usługi.

Przykład konfiguracji

Przegląd Ta sekcja opisuje prostą konfigurację monitorowania klastra wysokiej dostępności Zabbix jako usługi.

Wymagania wstępne Przed skonfigurowaniem monitorowania usług należy mieć skonfigurowane hosty:

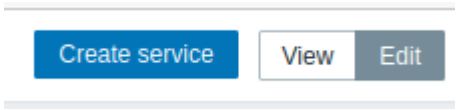
- *Węzeł HA 1* z co najmniej jednym wyzwalaczem i tagiem (najlepiej ustawionym na poziomie wyzwalacza) `component:ha-node-1`
- *Węzeł HA 2* z co najmniej jednym wyzwalaczem i tagiem (najlepiej ustawionym na poziomie wyzwalacza) `component:ha-node-2`

Drzewo usług Następnym krokiem jest zbudowanie drzewa usług. W tym przykładzie infrastruktura jest bardzo podstawowa i składa się z trzech usług: *Zabbix cluster* (nadrzędna) oraz dwóch usług podrzędnych *Zabbix server node 1* i *Zabbix server node 2*.

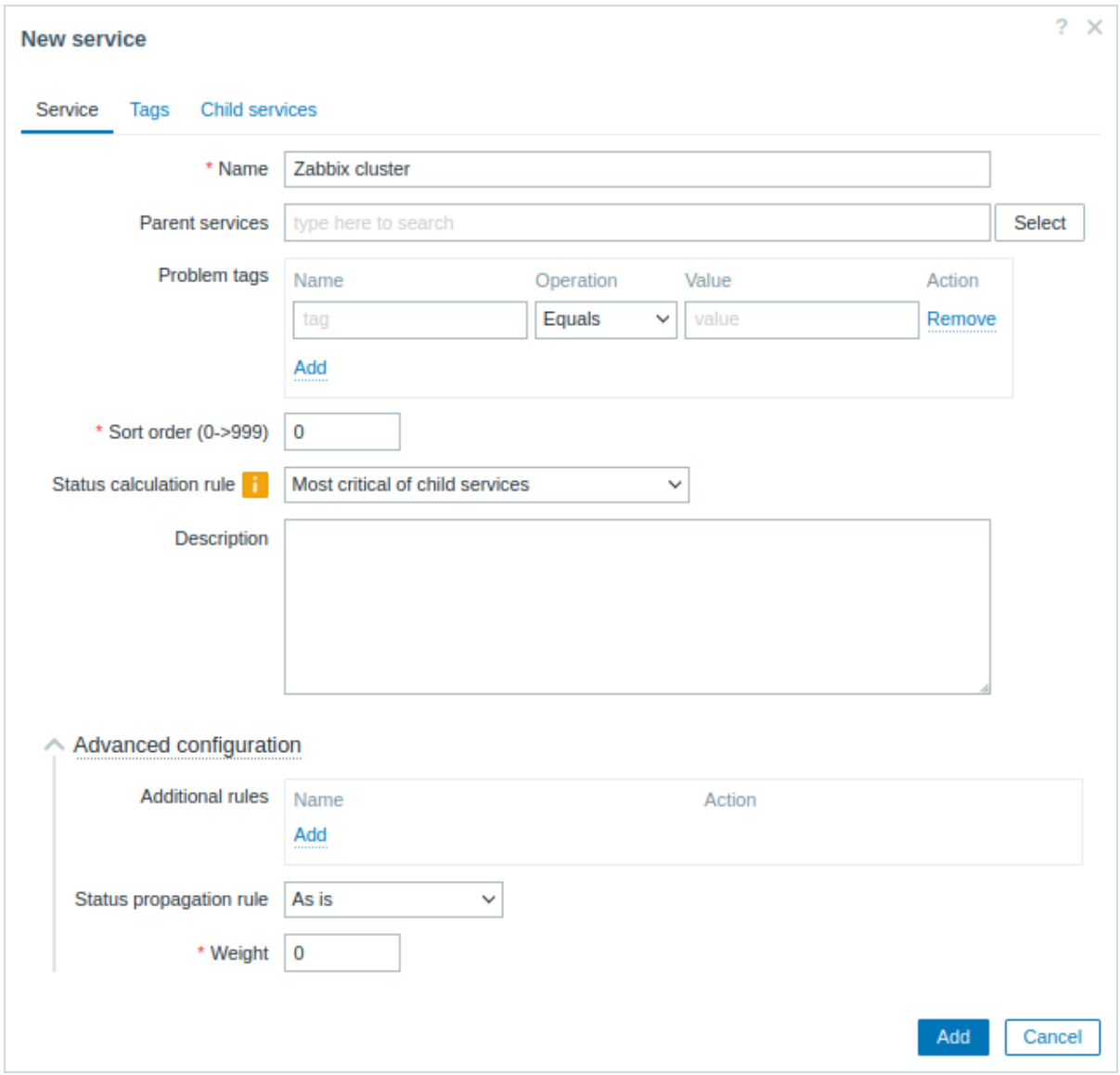
```
Zabbix cluster
|
```

```
|- Zabbix server node 1
|- Zabbix server node 2
```

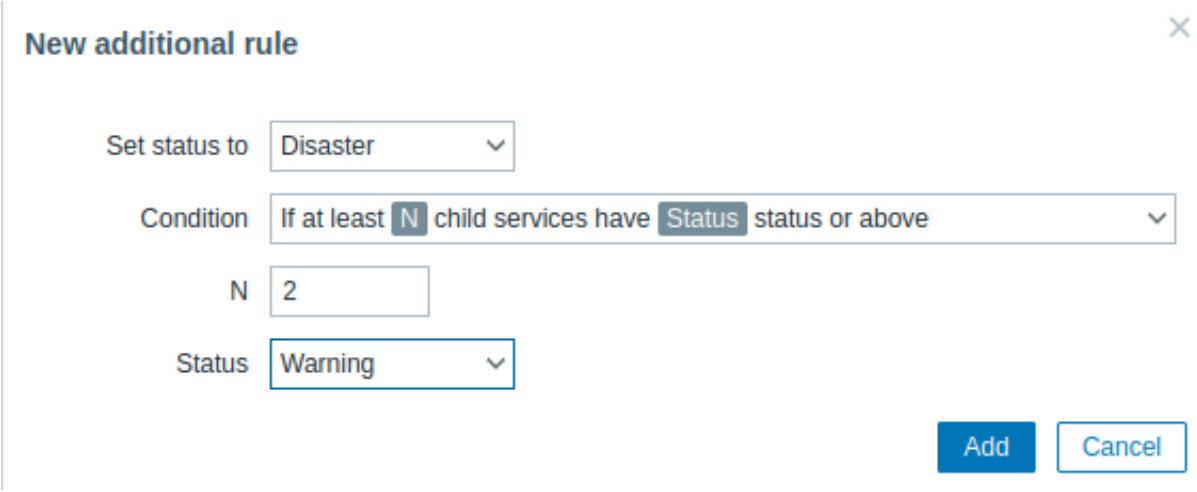
Na stronie Services włącz tryb *Edit* i kliknij *Create service*:



W oknie konfiguracji usługi wpisz nazwę *Zabbix cluster* i kliknij nagłówek *Advanced configuration*, aby wyświetlić zaawansowane opcje konfiguracji.



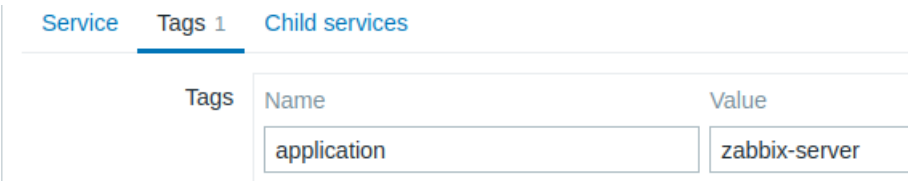
Skonfiguruj dodatkową regułę:



Zabbix cluster będzie mieć dwie usługi podrzędne - po jednej dla każdego z węzłów HA. Jeśli oba węzły HA mają problemy o statusie co najmniej *Warning*, status usługi nadrzędnej powinien zostać ustawiony na *Disaster*. Aby to osiągnąć, należy skonfigurować dodatkową regułę w następujący sposób:

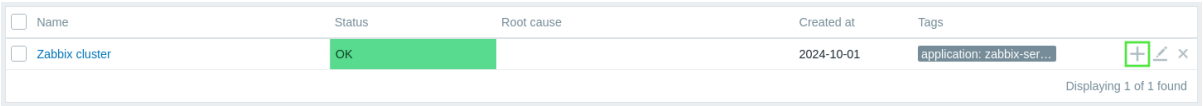
- Ustaw status na: Disaster
- Warunek: Jeśli co najmniej `N` usług podrzędnych ma status `Status` lub wyższy
- N: 2
- Status: Warning

Przejdź do zakładki *Tags* i dodaj tag `application:zabbix-server`. Ten tag będzie później używany dla akcji usług i raportów SLA.



Zapisz nową usługę.

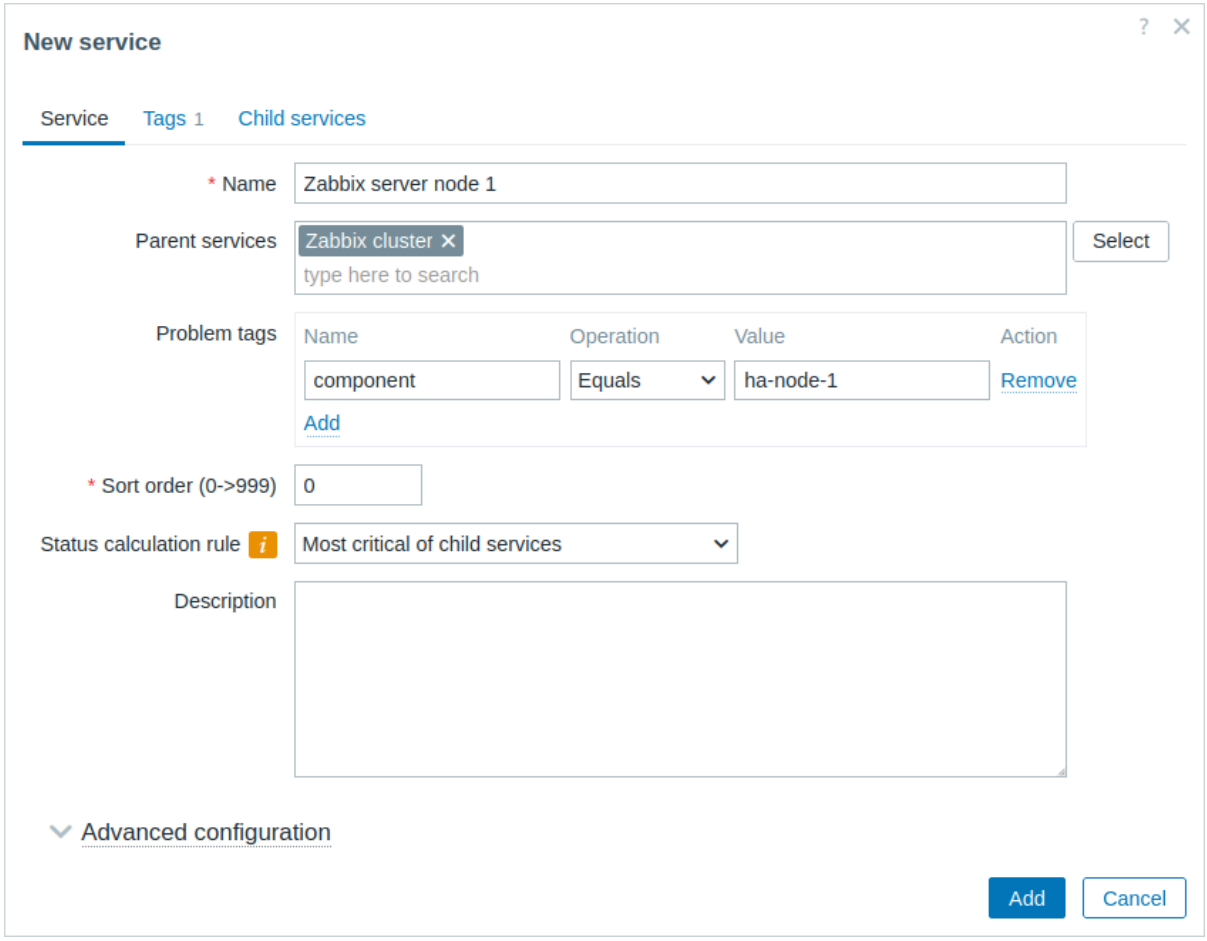
Aby dodać usługę podrzędną, kliknij ikonę plusa obok usługi klastra Zabbix (ikona jest widoczna tylko w trybie *Edit*).



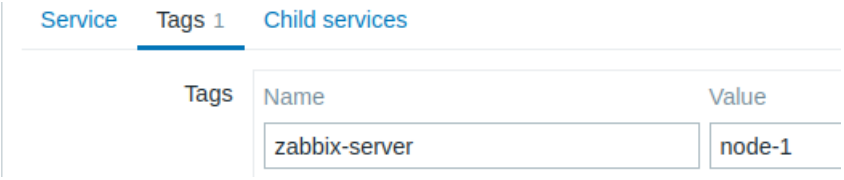
W oknie konfiguracji usługi wprowadź nazwę *Zabbix server node 1*. Zwróć uwagę, że parametr Parent services jest już wstępnie wypełniony wartością *Zabbix cluster*.

Dostępność tej usługi jest zależna od problemów na hoście *HA node 1*, oznaczonych tagiem problemu `component:ha-node-1`. W parametrze Problem tags wprowadź:

- Name: component
- Operation: Equals
- Value: ha-node-1



Przejdź do zakładki *Tags* i dodaj tag usługi: `zabbix-server:node-1`. Ten tag będzie później używany do akcji usług i raportów SLA.



Zapisz nową usługę.

Utwórz kolejną usługę podrzędną klastra Zabbix o nazwie „Zabbix server node 2”.

Ustaw tagi problemów w następujący sposób:

- Nazwa: component
- Operacja: Równa się
- Wartość: ha-node-2

Przejdź do zakładki *Tagi* i dodaj tag usługi: `zabbix-server:node-2`.

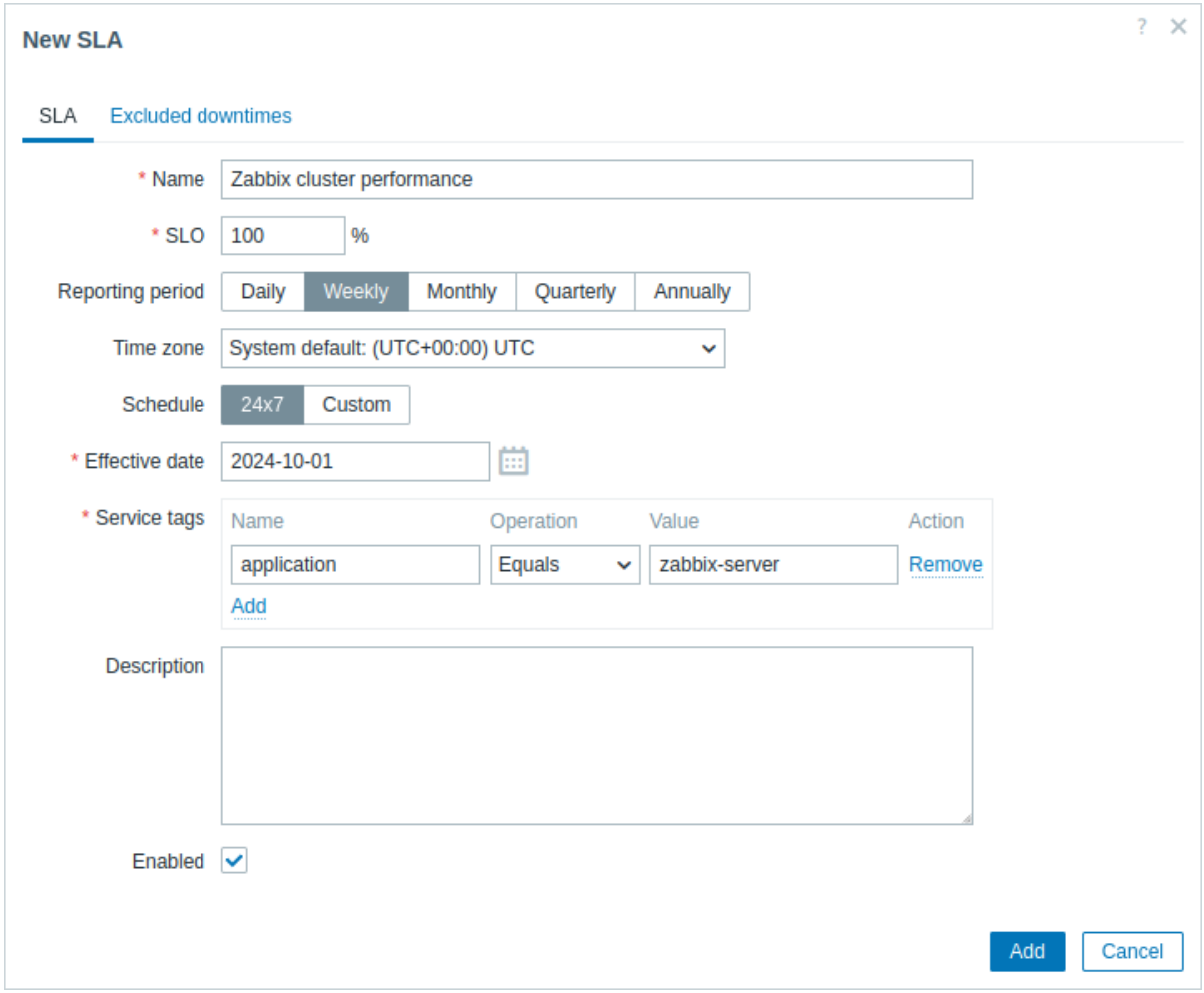
Zapisz nową usługę.

SLA W tym przykładzie oczekiwana wydajność klastra Zabbix wynosi 100%, z wyłączeniem półrocznego, jednogodzinnego okresu konserwacji.

Najpierw należy dodać nową umowę SLA:

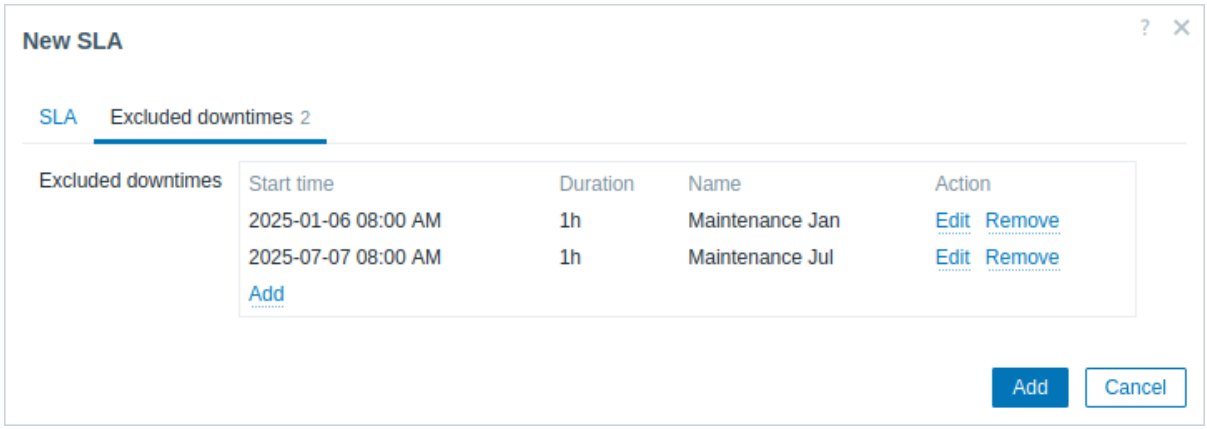
1. Przejdź do sekcji *Services* > *SLA*.
2. Kliknij *Create SLA*.
3. Wprowadź nazwę *Zabbix cluster performance* i ustaw SLO na 100%. Usługa Zabbix cluster ma znacznik usługi `application:zabbix-server`.
4. Aby używać tej umowy SLA do pomiaru wydajności klastra Zabbix, w parametrze *Service tags* określ:

- Name: application
- Operation: Equals
- Value: zabbix-server



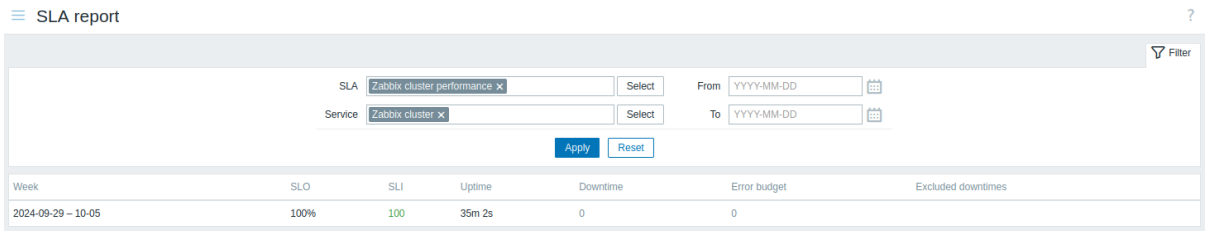
W rzeczywistym środowisku można również zaktualizować żądany okres raportowania, strefę czasową i datę rozpoczęcia albo zmienić harmonogram z 24/7 na niestandardowy. W tym przykładzie ustawienia domyślne są wystarczające.

5. Przejdź na kartę *Excluded downtimes* i dodaj okresy niedostępności dla zaplanowanych okresów konserwacji, aby wykluczyć je z obliczeń SLA.
6. W sekcji *Excluded downtimes* kliknij łącze *Add*, a następnie wprowadź nazwę okresu niedostępności, planowany czas rozpoczęcia i czas trwania.

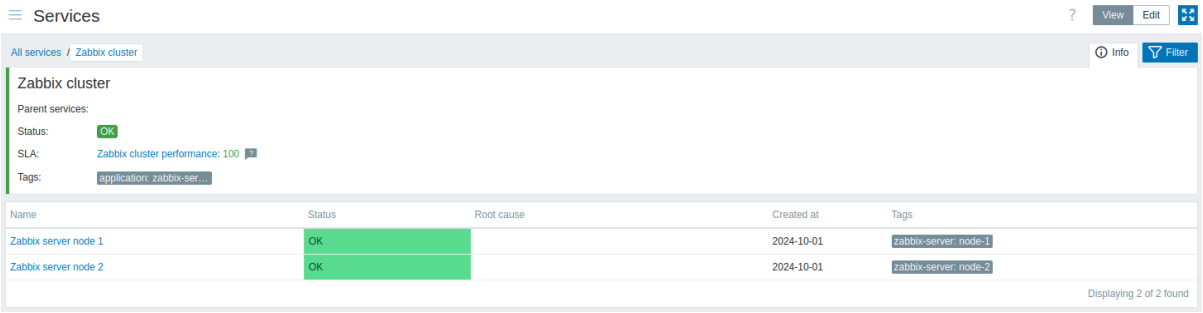


7. Kliknij *Add*, aby zapisać nową umowę SLA.

Przejdź do sekcji raportów SLA, aby wyświetlić raport SLA dla klastra Zabbix.



Informacje SLA można również sprawdzić w sekcji *Services*.



Ulepszenie monitorowania usług

Przegląd W Zabbix 6.0 funkcjonalność monitorowania usług została znacząco przebudowana (zobacz listę zmian w sekcji Co nowego w Zabbix 6.0.0).

Ta strona opisuje, w jaki sposób usługi i SLA, zdefiniowane we wcześniejszych wersjach Zabbix, są zmieniane podczas aktualizacji do Zabbix 6.0 lub nowszej.

Usługi W starszych wersjach Zabbixa usługi miały dwa typy zależności: miękkie i twarde. Po aktualizacji wszystkie zależności staną się równorzędne.

Jeśli usługa ”Child service” była wcześniej powiązana z ”Parent service 1” za pomocą zależności twardej, a dodatkowo z ”Parent service 2” za pomocą zależności miękkiej, po aktualizacji usługa ”Child service” będzie miała dwie usługi nadrzędne: ”Parent service 1” i ”Parent service 2”.

Mapowanie problemów i usług oparte na wyzwalaczach zostało zastąpione mapowaniem opartym na tagach. W Zabbixie 6.0 i nowszych formularz konfiguracji usługi ma nowy parametr *Tagi problemów*, który umożliwia określenie jednej lub wielu par nazwa tagu i wartość tagu do dopasowywania problemów. Wyzwalacze, które były powiązane z usługą, otrzymają nowy tag `ServiceLink : <trigger ID>:<trigger name>` (wartość tagu zostanie skrócona do 32 znaków). Powiązane usługi otrzymają `ServiceLink` tag problemu z tą samą wartością.

Zasady obliczania statusu

Algorytm obliczania statusu zostanie ulepszony zgodnie z następującymi zasadami:

- Nie obliczaj → Ustaw status na OK
- Problem, jeśli co najmniej jedno dziecko ma problem → Najbardziej krytyczny spośród usług podrzędnych
- Problem, jeśli wszystkie dzieci mają problemy → Najbardziej krytyczny, jeśli wszystkie dzieci mają problemy

SLA Wcześniej cele SLA musiały być definiowane osobno dla każdej usługi. Od Zabbix 6.0 SLA stało się osobnym bytem, który zawiera informacje o harmonogramie usługi, oczekiwanym poziomie realizacji usługi (SLO) oraz okresach przestoju, które mają być wyłączone z obliczeń. Po skonfigurowaniu SLA można przypisać do wielu usług za pomocą tagów usługi.

Podczas aktualizacji:

- Identyczne SLA zdefiniowane dla każdej usługi zostaną pogrupowane i dla każdej grupy zostanie utworzone jedno SLA.
- Każda objęta tym usługa otrzyma specjalny tag `SLA:<ID>`, a ten sam tag zostanie określony w parametrze *Service tags* odpowiedniego SLA.
- Czas utworzenia usługi, nowa metryka w raportach SLA, zostanie ustawiony na 01/01/2000 00:00 dla istniejących usług.

Szyfrowanie i bezpieczeństwo

Szyfrowanie Zabbix obsługuje szyfrowaną komunikację między komponentami Zabbix przy użyciu protokołu Transport Layer Security (TLS) w wersji 1.2 i 1.3 (w zależności od biblioteki kryptograficznej). Obsługiwane jest szyfrowanie oparte na certyfikatach oraz szyfrowanie oparte na kluczu współdzielonym.

Szyfrowanie można skonfigurować dla połączeń:

- Między serwerem Zabbix, proxy Zabbix, agentem Zabbix, usługą web Zabbix, narzędziami zabbix_sender i zabbix_get

- Do bazy danych Zabbix z frontend Zabbix i serwera/proxy
- Między frontend Zabbix i serwerem Zabbix

Szyfrowanie jest opcjonalne i można je skonfigurować dla poszczególnych komponentów:

- Niektóre proxy i agenty można skonfigurować tak, aby używały szyfrowania opartego na certyfikatach z serwerem, podczas gdy inne mogą używać szyfrowania opartego na kluczu współdzielonym, a jeszcze inne mogą nadal korzystać z nieszyfrowanej komunikacji (tak jak wcześniej).
- Serwer (proxy) może używać różnych konfiguracji szyfrowania dla różnych hostów.

Programy demonów Zabbix używają jednego portu nasłuchującego dla szyfrowanych i nieszyfrowanych połączeń przychodzących. Dodanie szyfrowania nie wymaga otwierania nowych portów na zaporach sieciowych.

Ograniczenia

- Klucze prywatne są przechowywane jako zwykły tekst w plikach czytelnych dla komponentów Zabbix podczas uruchamiania.
- Klucze współdzielone są wprowadzane w frontend Zabbix i przechowywane w bazie danych Zabbix jako zwykły tekst.
- Wbudowane szyfrowanie nie chroni komunikacji między serwerem WWW uruchamiającym frontend Zabbix a przeglądarką internetową użytkownika.
- Obecnie każde szyfrowane połączenie rozpoczyna się pełnym uzgadnianiem TLS; nie są zaimplementowane buforowanie sesji ani bilety sesji.
- Dodanie szyfrowania zwiększa czas sprawdzania pozycji i wykonywania akcji, w zależności od opóźnienia sieci:
  - Na przykład, jeśli opóźnienie pakietu wynosi 100 ms, otwarcie połączenia TCP i wysłanie nieszyfrowanego żądania zajmuje około 200 ms. Przy szyfrowaniu na ustanowienie połączenia TLS dodawane jest około 1000 ms.
  - Może być konieczne zwiększenie limitów czasu; w przeciwnym razie niektóre pozycje i akcje uruchamiające zdalne skrypty na agentach mogą działać przy nieszyfrowanych połączeniach, ale kończyć się przekroczeniem limitu czasu przy połączeniach szyfrowanych.
- Szyfrowanie nie jest obsługiwane przez odkrywanie sieci. Sprawdzenia agenta Zabbix wykonywane przez odkrywanie sieci będą nieszyfrowane i jeśli agent Zabbix jest skonfigurowany tak, aby odrzucać nieszyfrowane połączenia, takie sprawdzenia nie powiodą się.

Kompilowanie Zabbix z obsługą szyfrowania Aby obsługiwać szyfrowanie, Zabbix musi zostać skompilowany i połączony z jedną z obsługiwanych bibliotek kryptograficznych:

- GnuTLS - od wersji 3.1.18
- OpenSSL - wersje 1.0.1, 1.0.2, 1.1.0, 1.1.1, 3.0.x - 3.5.x
- LibreSSL - przetestowano z wersjami 2.7.4, 2.8.2:
  - LibreSSL 2.6.x nie jest obsługiwany
  - LibreSSL jest obsługiwany jako zgodny zamiennik OpenSSL; nowe funkcje API `tls_*()` specyficzne dla LibreSSL nie są używane. Komponenty Zabbix skompilowane z LibreSSL nie będą mogły używać PSK, można używać tylko certyfikatów.

Note:
Więcej informacji na temat konfigurowania SSL dla frontendu Zabbix można znaleźć w tych najlepszych praktykach.

Bibliotekę wybiera się, określając odpowiednią opcję w skrypcie „configure”:

- `--with-gnutls[=DIR]`
- `--with-openssl[=DIR]` (używane również dla LibreSSL)

Na przykład, aby skonfigurować źródła serwera i agenta z użyciem *OpenSSL*, można użyć polecenia:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --wit
```

Różne komponenty Zabbix mogą być kompilowane z użyciem różnych bibliotek kryptograficznych (np. serwer z *OpenSSL*, a agent z *GnuTLS*).

Attention:
Jeśli planujesz używać kluczy wstępnie współdzielonych (PSK), rozważ użycie bibliotek *GnuTLS* lub *OpenSSL 1.1.0* (albo nowszych) w komponentach Zabbix używających PSK. Biblioteki *GnuTLS* i *OpenSSL 1.1.0* obsługują zestawy szyfrów PSK z utajnieniem przekazywania doskonałym. Starsze wersje biblioteki *OpenSSL* (1.0.1, 1.0.2c) również obsługują PSK, ale dostępne zestawy szyfrów PSK nie zapewniają utajnienia przekazywania doskonałego.

Zarządzanie szyfrowaniem połączeń Połączenia w Zabbix mogą używać:

- braku szyfrowania (domyślnie)
- szyfrowania opartego na certyfikatach RSA
- szyfrowania opartego na PSK

Do określania szyfrowania między komponentami Zabbix służą dwa ważne parametry:

- TLSConnect - określa szyfrowanie używane dla połączeń wychodzących (nieszyfrowane, PSK lub certyfikat)
- TLSAccept - określa typy połączeń dozwolonych dla połączeń przychodzących (nieszyfrowane, PSK lub certyfikat). Można określić jedną lub więcej wartości.

`TLSConnect` jest używany w plikach konfiguracyjnych Zabbix proxy (w trybie aktywnym określa tylko połączenia z serwerem) oraz Zabbix agent (dla aktywnych kontroli). W frontend Zabbix odpowiednikiem `TLSConnect` jest pole *Połączenia z hostem* w sekcji *Zbieranie danych* > *Hosty* > <some host> > karta *Szyfrowanie* oraz pole *Połączenia z proxy* w sekcji *Administracja* > *Proxy* > <some proxy> > karta *Szyfrowanie*. Jeśli skonfigurowany typ szyfrowania połączenia nie powiedzie się, nie zostaną podjęte próby użycia innych typów szyfrowania.

`TLSAccept` jest używany w plikach konfiguracyjnych Zabbix proxy (w trybie pasywnym określa tylko połączenia z serwera) oraz Zabbix agent (dla pasywnych kontroli). W frontend Zabbix odpowiednikiem `TLSAccept` jest pole *Połączenia z hosta* w sekcji *Zbieranie danych* > *Hosty* > <some host> > karta *Szyfrowanie* oraz pole *Połączenia z proxy* w sekcji *Administracja* > *Proxy* > <some proxy> > karta *Szyfrowanie*.

Zwykle konfiguruje się tylko jeden typ szyfrowania dla połączeń przychodzących. Można jednak chcieć zmienić typ szyfrowania, np. z nieszyfrowanego na oparte na certyfikatach, przy minimalnym przestoju i możliwości wycofania zmian. Aby to osiągnąć:

- Ustaw `TLSAccept=unencrypted,cert` w pliku konfiguracyjnym agenta i uruchom ponownie Zabbix agent.
- Przetestuj połączenie z agentem za pomocą `zabbix_get`, używając certyfikatu. Jeśli działa, możesz ponownie skonfigurować szyfrowanie dla tego agenta w frontend Zabbix, w sekcji *Zbieranie danych* > *Hosty* > <some host> > karta *Szyfrowanie*, ustawiając *Połączenia z hostem* na *Certyfikat*.
- Po zaktualizowaniu pamięci podręcznej konfiguracji serwera (oraz zaktualizowaniu konfiguracji proxy, jeśli host jest monitorowany przez proxy) połączenia z tym agentem będą szyfrowane.
- Jeśli wszystko działa zgodnie z oczekiwaniami, możesz ustawić `TLSAccept=cert` w pliku konfiguracyjnym agenta i uruchomić ponownie Zabbix agent. Agent będzie teraz akceptował wyłącznie szyfrowane połączenia oparte na certyfikatach. Połączenia nieszyfrowane i oparte na PSK będą odrzucane.

W podobny sposób działa to na serwerze i proxy. Jeśli w konfiguracji hosta w frontend Zabbix opcja *Połączenia z hosta* jest ustawiona na „Certyfikat", od agenta (aktywne kontrole) i zabbix_sender (pozycje typu trapper) będą akceptowane wyłącznie szyfrowane połączenia oparte na certyfikatach.

Najprawdopodobniej skonfigurujesz połączenia przychodzące i wychodzące tak, aby używały tego samego typu szyfrowania albo w ogóle nie były szyfrowane. Technicznie możliwe jest jednak skonfigurowanie ich asymetrycznie, np. szyfrowania opartego na certyfikatach dla połączeń przychodzących i opartego na PSK dla połączeń wychodzących.

Konfiguracja szyfrowania dla każdego hosta jest wyświetlana w frontend Zabbix, w sekcji *Zbieranie danych* > *Hosty*, w kolumnie *Szyfrowanie agenta*. Na przykład:

Przykład	Połączenia z hostem	Dozwolone połączenia z hosta	Odrzucone połączenia z hosta
NONE	Nieszyfrowane	Nieszyfrowane	Szyfrowane, oparte na certyfikatach i PSK
CERT NONE PSK CERT	Szyfrowane, oparte na certyfikatach	Szyfrowane, oparte na certyfikatach	Nieszyfrowane i szyfrowane, oparte na PSK
PSK NONE PSK CERT	Szyfrowane, oparte na PSK	Szyfrowane, oparte na PSK	Nieszyfrowane i szyfrowane, oparte na certyfikatach
PSK NONE PSK CERT	Szyfrowane, oparte na PSK	Nieszyfrowane i szyfrowane, oparte na PSK	Szyfrowane, oparte na certyfikatach
CERT NONE PSK CERT	Szyfrowane, oparte na certyfikatach	Nieszyfrowane, szyfrowane oparte na PSK lub certyfikatach	-

Attention:
Połączenia są domyślnie nieszyfrowane. Szyfrowanie należy skonfigurować osobno dla każdego hosta i proxy.

zabbix_get i zabbix_sender z szyfrowaniem Zobacz strony podręcznika zabbix_get i zabbix_sender dotyczące używania ich z szyfrowaniem.

Zestawy szyfrów Zestawy szyfrów są domyślnie konfigurowane wewnętrznie podczas uruchamiania Zabbixa.

Obsługiwane są również zestawy szyfrów skonfigurowane przez użytkownika dla GnuTLS i OpenSSL. Użytkownicy mogą skonfigurować zestawy szyfrów zgodnie z własnymi zasadami bezpieczeństwa. Korzystanie z tej funkcji jest opcjonalne (wbudowane domyślne zestawy szyfrów nadal działają).

W przypadku bibliotek kryptograficznych skompilowanych z ustawieniami domyślnymi wbudowane reguły Zabbixa zwykle dają następujące zestawy szyfrów (w kolejności od wyższego do niższego priorytetu):

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
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

Ciphersuites zdefiniowane przez użytkownika Wbudowane kryteria wyboru ciphersuite można zastąpić za pomocą ciphersuite zdefiniowanych przez użytkownika.

Attention:
Ciphersuite zdefiniowane przez użytkownika to funkcja przeznaczona dla zaawansowanych użytkowników, którzy rozumieją ciphersuite TLS, ich bezpieczeństwo i konsekwencje błędów, oraz którzy czują się swobodnie podczas rozwiązywania problemów z TLS.

Wbudowane kryteria wyboru ciphersuite można zastąpić przy użyciu następujących parametrów:

Zakres nadpisania	Parametr	Wartość	Opis
Wybór ciphersuite dla certyfikatów	TLSCipherCert13	Prawidłowe ciągi OpenSSL 1.1.1 cipher strings dla protokołu TLS 1.3 (ich wartości są przekazywane do funkcji OpenSSL `SSL_CTX_set_ciphersuites()`).	Kryteria wyboru ciphersuite oparte na certyfikatach dla TLS 1.3 Tylko OpenSSL 1.1.1 lub nowszy.
	TLSCipherCert	Prawidłowe ciągi OpenSSL cipher strings dla TLS 1.2 lub prawidłowe ciągi priorytetów GnuTLS priority strings. Ich wartości są przekazywane odpowiednio do funkcji `SSL_CTX_set_cipher_list()` lub `gnutls_priority_init()`.	Kryteria wyboru ciphersuite oparte na certyfikatach dla TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)

Zakres nadpisania	Parametr	Wartość	Opis
Wybór ciphersuite dla PSK	TLSCipherPSK13	Prawidłowe ciągi OpenSSL 1.1.1 cipher strings dla protokołu TLS 1.3 (ich wartości są przekazywane do funkcji OpenSSL `SSL_CTX_set_ciphersuites()`).	Kryteria wyboru ciphersuite oparte na PSK dla TLS 1.3 Tylko OpenSSL 1.1.1 lub nowszy.
	TLSCipherPSK	Prawidłowe ciągi OpenSSL cipher strings dla TLS 1.2 lub prawidłowe ciągi priorytetów GnuTLS priority strings. Ich wartości są przekazywane odpowiednio do funkcji `SSL_CTX_set_cipher_list()` lub `gnutls_priority_init()`.	Kryteria wyboru ciphersuite oparte na PSK dla TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)
Łączna lista ciphersuite dla certyfikatu i PSK	TLSCipherAll13	Prawidłowe ciągi OpenSSL 1.1.1 cipher strings dla protokołu TLS 1.3 (ich wartości są przekazywane do funkcji OpenSSL `SSL_CTX_set_ciphersuites()`).	Kryteria wyboru ciphersuite dla TLS 1.3 Tylko OpenSSL 1.1.1 lub nowszy.
	TLSCipherAll	Prawidłowe ciągi OpenSSL cipher strings dla TLS 1.2 lub prawidłowe ciągi priorytetów GnuTLS priority strings. Ich wartości są przekazywane odpowiednio do funkcji `SSL_CTX_set_cipher_list()` lub `gnutls_priority_init()`.	Kryteria wyboru ciphersuite dla TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)

Aby nadpisać wybór ciphersuite w narzędziach zabbix_get i zabbix_sender, użyj parametrów wiersza poleceń:

- `--tls-cipher13`
- `--tls-cipher`

Nowe parametry są opcjonalne. Jeśli parametr nie zostanie określony, używana jest wewnętrzna wartość domyślna. Jeśli parametr jest zdefiniowany, nie może być pusty.

Jeśli ustawienie wartości TLSCipher* w bibliotece kryptograficznej zakończy się niepowodzeniem, to serwer, proxy lub agent nie uruchomi się i zostanie zapisany błąd.

Ważne jest, aby zrozumieć, kiedy każdy z parametrów ma zastosowanie.

Połączenia wychodzące

Najprostszym przypadkiem są połączenia wychodzące:

- W przypadku połączeń wychodzących z certyfikatem - użyj TLSCipherCert13 lub TLSCipherCert
- W przypadku połączeń wychodzących z PSK - użyj TLSCipherPSK13 lub TLSCipherPSK
- W przypadku narzędzi zabbix_get i zabbix_sender można użyć parametrów wiersza poleceń `--tls-cipher13` lub `--tls-cipher` (szyfrowanie jest jednoznacznie określone za pomocą parametru `--tls-connect`)

Połączenia przychodzące

W przypadku połączeń przychodzących jest to nieco bardziej skomplikowane, ponieważ reguły są specyficzne dla komponentów i konfiguracji.

Dla Zabbix **agent**:

Konfiguracja połączenia agenta	Konfiguracja szyfrów
TLSConnect=cert	TLSCipherCert, TLSCipherCert13
TLSConnect=psk	TLSCipherPSK, TLSCipherPSK13
TLSAccept=cert	TLSCipherCert, TLSCipherCert13
TLSAccept=psk	TLSCipherPSK, TLSCipherPSK13
TLSAccept=cert,psk	TLSCipherAll, TLSCipherAll13

Dla Zabbix **serwer** i **proxy**:

Konfiguracja połączenia	Konfiguracja szyfrów
Połączenia wychodzące z użyciem PSK	TLSCipherPSK, TLSCipherPSK13
Połączenia przychodzące z użyciem certyfikatów	TLSCipherAll, TLSCipherAll13

Konfiguracja połączenia	Konfiguracja szyfrów
Połączenia przychodzące z użyciem PSK, jeśli serwer nie ma certyfikatu	TLSCipherPSK, TLSCipherPSK13
Połączenia przychodzące z użyciem PSK, jeśli serwer ma certyfikat	TLSCipherAll, TLSCipherAll13

W dwóch powyższych tabelach można zauważyć pewien wzorzec:

- TLSCipherAll i TLSCipherAll13 można określić tylko wtedy, gdy używana jest łączna lista zestawów szyfrów opartych na certyfikatach **i** PSK. Występują dwa przypadki, gdy ma to miejsce: serwer (proxy) z skonfigurowanym certyfikatem (zestawy szyfrów PSK są zawsze skonfigurowane na serwerze, proxy jeśli biblioteka kryptograficzna obsługuje PSK), agent skonfigurowany do akceptowania zarówno przychodzących połączeń opartych na certyfikatach, jak i na PSK
- w pozostałych przypadkach wystarczające są TLSCipherCert* i/lub TLSCipherPSK*

Poniższe tabele pokazują wbudowane domyślne wartości `TLSCipher*`. Mogą one stanowić dobry punkt wyjścia do własnych niestandardowych wartości.

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] Domyślne wartości są inne dla starszych wersji OpenSSL (1.0.1, 1.0.2, 1.1.0), dla LibreSSL oraz jeśli OpenSSL został skompilowany bez obsługi PSK.

Przykłady zestawów szyfrów skonfigurowanych przez użytkownika

Poniżej znajdują się następujące przykłady zestawów szyfrów skonfigurowanych przez użytkownika:

- Testowanie ciągów szyfrów i zezwalanie wyłącznie na zestawy szyfrów PFS
- Przełączanie z AES128 na AES256

Testowanie ciągów szyfrów i zezwalanie wyłącznie na zestawy szyfrów PFS

Aby sprawdzić, które zestawy szyfrów zostały wybrane, należy ustawić `DebugLevel=4` w pliku konfiguracyjnym lub użyć opcji `-vv` dla zabbix_sender.

Przed uzyskaniem żądanych zestawów szyfrów może być konieczne przeprowadzenie kilku eksperymentów z parametrami `TLSCipher*`. Wielokrotne ponowne uruchamianie serwera Zabbix, proxy lub agenta tylko po to, aby dostosować parametry `TLSCipher*`, jest niewygodne. Wygodniej jest użyć zabbix_sender lub polecenia `openssl`. Pokażmy obie metody.

1. Użycie zabbix_sender.

Utwórzmy testowy plik konfiguracyjny, na przykład `/home/zabbix/test.conf`, używając składni pliku `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
```

W tym przykładzie potrzebne są prawidłowe certyfikaty CA i agenta oraz PSK. Dostosuj ścieżki i nazwy plików certyfikatów i PSK do swojego środowiska.

Jeśli nie używasz certyfikatów, a tylko PSK, możesz utworzyć prostszy plik testowy:

```
Hostname=nonexisting
ServerActive=nonexisting

TLSConnect=psk
TLSPSKIdentity=nonexisting
TLSPSKFile=/home/zabbix/agentd.psk
```

Wybrane zestawy szyfrów można wyświetlić, uruchamiając zabbix_sender (przykład skompilowany z 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
```

Tutaj widoczne są domyślnie wybrane zestawy szyfrów. Te wartości domyślne zostały dobrane w celu zapewnienia interoperacyjności z agentami Zabbix działającymi w systemach ze starszymi wersjami OpenSSL (od 1.0.1).

W nowszych systemach można zwiększyć bezpieczeństwo, zezwalając tylko na kilka zestawów szyfrów, na przykład wyłącznie na zestawy szyfrów z PFS (Perfect Forward Secrecy). Skonfigurujmy parametry TLSCipher*, aby zezwalały tylko na zestawy szyfrów obsługujące PFS.

Attention:
Wynik nie będzie interoperacyjny z systemami używającymi OpenSSL 1.0.1 i 1.0.2, jeśli używany jest PSK. Szyfrowanie oparte na certyfikatach powinno działać.

Dodaj dwa wiersze do pliku konfiguracyjnego `test.conf`:

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
```

i ponownie wykonaj test:

```
$ 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
```

Listy „certificate ciphersuites" i „PSK ciphersuites" uległy zmianie - są krótsze niż wcześniej i zawierają tylko zestawy szyfrów TLS 1.3 oraz zestawy szyfrów TLS 1.2 ECDHE-*, zgodnie z oczekiwaniami.

2. Parametrów TLSCipherAll i TLSCipherAll13 nie można testować za pomocą zabbix_sender; nie wpływają one na wartość „certificate and PSK ciphersuites" pokazaną w powyższym przykładzie. Aby dostosować TLSCipherAll i TLSCipherAll13, należy przeprowadzić eksperymenty z agentem, proxy lub serwerem.

Aby zezwalać wyłącznie na zestawy szyfrów PFS, może być konieczne dodanie maksymalnie trzech parametrów

```
TLSCipherCert=EECDH+aRSA+AES128
TLSCipherPSK=kECDHEPSK+AES128
TLSCipherAll=EECDH+aRSA+AES128:kECDHEPSK+AES128
```

do plików `zabbix_agentd.conf`, `zabbix_proxy.conf` i `zabbix_server.conf`, jeśli każdy z nich ma skonfigurowany certyfikat, a agent ma również skonfigurowany PSK.

Jeśli środowisko Zabbix używa wyłącznie szyfrowania opartego na PSK i nie używa certyfikatów, wystarczy tylko jeden parametr:

```
TLSCipherPSK=kECDHEPSK+AES128
```

Teraz, gdy wiesz już, jak to działa, możesz testować wybór zestawów szyfrów również poza Zabbix, za pomocą polecenia `openssl`. Przetestujmy wartości wszystkich trzech parametrów 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
```

Możesz preferować użycie `openssl ciphers` z opcją `-V`, aby uzyskać bardziej szczegółowe dane wyjściowe:

```
$ 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
```

Podobnie można testować ciągi priorytetów dla 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
```

Przełączanie z AES128 na AES256

Zabbix używa AES128 jako wbudowanego domyślnego algorytmu dla danych. Załóżmy, że używasz certyfikatów i chcesz przełączyć się na AES256 w OpenSSL 1.1.1.

Można to osiągnąć, dodając odpowiednie parametry w pliku `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:
Chociaż będą używane wyłącznie zestawy szyfrów związane z certyfikatami, parametry `TLSCipherPSK*` są również zdefiniowane, aby uniknąć ich wartości domyślnych, które obejmują mniej bezpieczne szyfry dla szerszej interoperacyjności. Zestawów szyfrów PSK nie można całkowicie wyłączyć na serwerze/proxy.

A w pliku `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
```

Rozwiązywanie problemów z szyfrowaniem Ogólne zalecenia:

- Zacznij od ustalenia, który komponent działa jako klient TLS, a który jako serwer TLS w danym przypadku problemowym.
 serwer Zabbix, proxy i agenty, w zależności od interakcji między nimi, mogą działać zarówno jako serwery TLS, jak i klienci TLS.
 Na przykład serwer Zabbix łączący się z agentem w celu wykonania pasywnej kontroli działa jako klient TLS. Agent pełni rolę serwera TLS.
 Agent Zabbix, pobierający listę aktywnych kontroli z proxy, działa jako klient TLS. Proxy pełni rolę serwera TLS.
 Narzędzia `zabbix_get` i `zabbix_sender` zawsze działają jako klienci TLS.
- Zabbix używa uwierzytelniania wzajemnego.
 Każda ze stron weryfikuje swój odpowiednik i może odrzucić połączenie.
 Na przykład serwer Zabbix łączący się z agentem może natychmiast zamknąć połączenie, jeśli certyfikat agenta

jest nieprawidłowy. I odwrotnie - agent Zabbix przyjmujący połączenie z serwera może zamknąć połączenie, jeśli serwer nie jest zaufany przez agenta.

- Sprawdź pliki dziennika po obu stronach - po stronie klienta TLS i serwera TLS.
 Strona, która odrzuca połączenie, może zapisać dokładny powód odrzucenia. Druga strona często zgłasza raczej ogólny błąd (np. "Connection closed by peer", "connection was non-properly terminated").
- Czasami nieprawidłowo skonfigurowane szyfrowanie skutkuje mylącymi komunikatami o błędach, które w żaden sposób nie wskazują rzeczywistej przyczyny.
 W poniższych podsekcjach staramy się przedstawić (daleko niepełną) listę komunikatów i możliwych przyczyn, które mogą pomóc w rozwiązywaniu problemów.
 Należy pamiętać, że różne zestawy narzędzi kryptograficznych (OpenSSL, GnuTLS) często generują różne komunikaty o błędach w tych samych sytuacjach problemowych.
 Czasami komunikaty o błędach zależą nawet od konkretnej kombinacji zestawów narzędzi kryptograficznych po obu stronach.

Rozwiązywanie problemów z typem połączenia lub uprawnieniami Serwer jest skonfigurowany do łączenia się z agentem przy użyciu PSK, ale agent akceptuje tylko nieszyfrowane połączenia

W logu serwera lub proxy (z *GnuTLS* 3.3.16)

```
Pobranie wartości z agenta nie powiodło się: zbx_tls_connect(): gnutls_handshake() failed: \
    -110 Połączenie TLS zostało nieprawidłowo zakończone.
```

W logu serwera lub proxy (z *OpenSSL* 1.0.2c)

```
Pobranie wartości z agenta nie powiodło się: Połączenie TCP zostało nawiązane pomyślnie, nie można ustanow
    Połączenie zamknięte przez drugą stronę. Sprawdź dozwolone typy połączeń i prawa dostępu
```

Jedna strona łączy się z certyfikatem, ale druga strona akceptuje tylko PSK lub odwrotnie

W dowolnym logu (z *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.
```

W dowolnym logu (z *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"
```

Próba użycia Zabbix sender skompilowanego z obsługą TLS do wysłania danych do serwera/proxy Zabbix skompilowanego bez obsługi TLS W logu po stronie łączącej:

Linux:

```
...W zbx_tls_init_child()
...Biblioteka OpenSSL (wersja OpenSSL 1.1.1  11 Sep 2018) zainicjalizowana
...
...W zbx_tls_connect(): psk_identity:"PSK test sender"
...Koniec zbx_tls_connect():FAIL błąd:'połączenie zamknięte przez drugą stronę'
...błąd wysyłania wartości: TCP zakończone powodzeniem, nie można ustanowić TLS do [[localhost]:10051]: po
```

Windows:

```
...Biblioteka OpenSSL (wersja OpenSSL 1.1.1a  20 Nov 2018) zainicjalizowana
...
...W zbx_tls_connect(): psk_identity:"PSK test sender"
...zbx_psk_client_cb() zażądał tożsamości PSK "PSK test sender"
...Koniec zbx_tls_connect():FAIL błąd:'SSL_connect() błąd I/O: [0x00000000] Operacja została pomyślnie zak
...błąd wysyłania wartości: TCP zakończone powodzeniem, nie można ustanowić TLS do [[192.168.1.2]:10051]:
```

W dzienniku po stronie odbierającej:

```
...nie udało się zaakceptować połączenia przychodzącego: z 127.0.0.1: obsługa TLS nie została skompilowana
```

Jedna strona łączy się z PSK, ale druga strona używa LibreSSL lub została skompilowana bez obsługi szyfrowania LibreSSL nie obsługuje PSK.

W logu strony łączącej się:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() I/O error: [0] Success
```

W logu strony przyjmującej połączenie:

```
...failed to accept an incoming connection: from 192.168.1.2: support for PSK was not compiled in
```

W frontend Zabbix:

```
Get value from agent failed: TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect()
```

Jedna strona łączy się z PSK, ale druga strona używa OpenSSL z wyłączonym wsparciem PSK W logu strony łączącej:

```
...TCP successful, cannot establish TLS to [[192.168.1.2]:10050]: SSL_connect() set result code to SSL_ERR
```

W logu strony przyjmującej:

```
...failed to accept an incoming connection: from 192.168.1.2: TLS handshake set result code to 1: file ssl
```

Szyfrowanie przy użyciu certyfikatów

Omówienie Zabbix może używać certyfikatów RSA w formacie PEM, podpisanych przez publiczny lub wewnętrzny urząd certyfikacji (CA).

Weryfikacja certyfikatu jest wykonywana względem wstępnie skonfigurowanego certyfikatu CA. Opcjonalnie można używać list odwołanych certyfikatów (CRL).

Każdy komponent Zabbix może mieć skonfigurowany tylko jeden certyfikat.

Więcej informacji na temat konfiguracji i obsługi wewnętrznego CA, generowania i podpisywania żądań certyfikatów oraz unieważniania certyfikatów można znaleźć w samouczkach, takich jak OpenSSL PKI Tutorial v2.0.

Należy dokładnie rozważyć i przetestować rozszerzenia certyfikatu. Więcej informacji można znaleźć w sekcji Ograniczenia dotyczące używania rozszerzeń certyfikatu X.509 v3.

Parametry konfiguracji certyfikatów Poniższe parametry konfiguracji są obsługiwane podczas konfigurowania certyfikatów na komponentach Zabbix.

Parameter	Mandatory	Description
TLSCAFile	yes	Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnego urzędu CA do weryfikacji certyfikatu drugiej strony. Jeśli używasz łańcucha certyfikatów z wieloma elementami, uporządkuj certyfikaty tak, aby najpierw znajdowały się certyfikaty CA niższego poziomu, a następnie certyfikaty CA wyższego poziomu. Certyfikaty z wielu urzędów CA mogą być umieszczone w jednym pliku.
TLSCRLFile	no	Pełna ścieżka do pliku zawierającego listy unieważnionych certyfikatów (CRL).
TLSCertFile	yes	Pełna ścieżka do pliku zawierającego certyfikat. Jeśli używasz łańcucha certyfikatów z wieloma elementami, uporządkuj certyfikaty tak, aby najpierw znajdował się certyfikat serwera, proxy lub agenta, następnie certyfikaty CA niższego poziomu, a na końcu certyfikaty CA wyższego poziomu.
TLSKeyFile	yes	Pełna ścieżka do pliku zawierającego klucz prywatny. Upewnij się, że ten plik jest czytelny wyłącznie dla użytkownika Zabbix, ustawiając odpowiednie prawa dostępu.
TLSServerCertIssuer	no	Dozwolony wystawca certyfikatu serwera.
TLSServerCertSubject	no	Dozwolony podmiot certyfikatu serwera.

Przykłady konfiguracji Po skonfigurowaniu niezbędnych certyfikatów skonfiguruj komponenty Zabbix do korzystania z szyfrowania opartego na certyfikatach.

Poniżej znajdują się szczegółowe kroki konfiguracji:

- serwer Zabbix
- proxy Zabbix
- agent Zabbix

serwer Zabbix

1. Przygotuj plik certyfikatu CA.

Aby weryfikować certyfikaty równorzędnych stron, serwer Zabbix musi mieć dostęp do pliku zawierającego certyfikaty głównych, samopodpisanych urzędów certyfikacji CA najwyższego poziomu. Na przykład, jeśli potrzebne są certyfikaty z dwóch niezależnych głównych urzędów CA, umieść je w pliku `/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
....
9wEzdN8uTrqoyU78gi12npLj08LegRKjb5hFTVm0
-----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. Umieść certyfikat lub łańcuch certyfikatów serwera Zabbix w pliku, na przykład `/home/zabbix/zabbix_server.crt`. Pierwszym certyfikatem jest certyfikat serwera Zabbix, a po nim znajduje się certyfikat pośredniego urzędu CA:

```
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/wUtcVIvWXdMIFJM0Hw==
-----END CERTIFICATE-----
```

Note:
Używaj wyłącznie atrybutów wymienionych powyżej zarówno dla certyfikatów klienta, jak i serwera, aby uniknąć wpływu na proces weryfikacji certyfikatu. Na przykład OpenSSL może nie nawiązać zaszyfrowanego połączenia, jeśli zostaną użyte rozszerzenia *X509v3 Subject Alternative Name* lub *Netscape Cert Type*. Więcej informacji można znaleźć w sekcji Ograniczenia dotyczące używania rozszerzeń certyfikatów X.509 v3.

3. Umieść klucz prywatny serwera Zabbix w pliku, na przykład `/home/zabbix/zabbix_server.key`:

```
-----BEGIN PRIVATE KEY-----
MIIEwAIBADANBgkqhkiG9w0BAQEFAASCBKowggSmAgEAAoIBAQC9tIXIJoVnNXDl
...
IJLkhbybBYEf47MLhffWa7XvZTY=
-----END PRIVATE KEY-----
```

4. Edytuj parametry konfiguracji TLS w pliku konfiguracyjnym serwera Zabbix:

```
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_server.crt
TLSKeyFile=/home/zabbix/zabbix_server.key
```

Zabbix proxy

1. Przygotuj pliki zawierające certyfikaty głównego urzędu certyfikacji (CA), certyfikat lub łańcuch certyfikatów Zabbix proxy oraz klucz prywatny, jak opisano w sekcji Zabbix server. Następnie odpowiednio edytuj parametry `TLSCAFile`, `TLSCertFile` i `TLSKeyFile` w pliku konfiguracyjnym proxy Zabbix.

2. Edytuj dodatkowe parametry TLS w pliku konfiguracyjnym proxy Zabbix:

- Dla proxy aktywnego: `TLSConnect=cert`
- Dla proxy pasywnego: `TLSAccept=cert`

Note:
Aby zwiększyć bezpieczeństwo proxy, możesz również ustawić parametry `TLSServerCertIssuer` i `TLSServerCertSubject`. Więcej informacji znajdziesz w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.

Parametry TLS w końcowym pliku konfiguracyjnym proxy mogą wyglądać następująco:

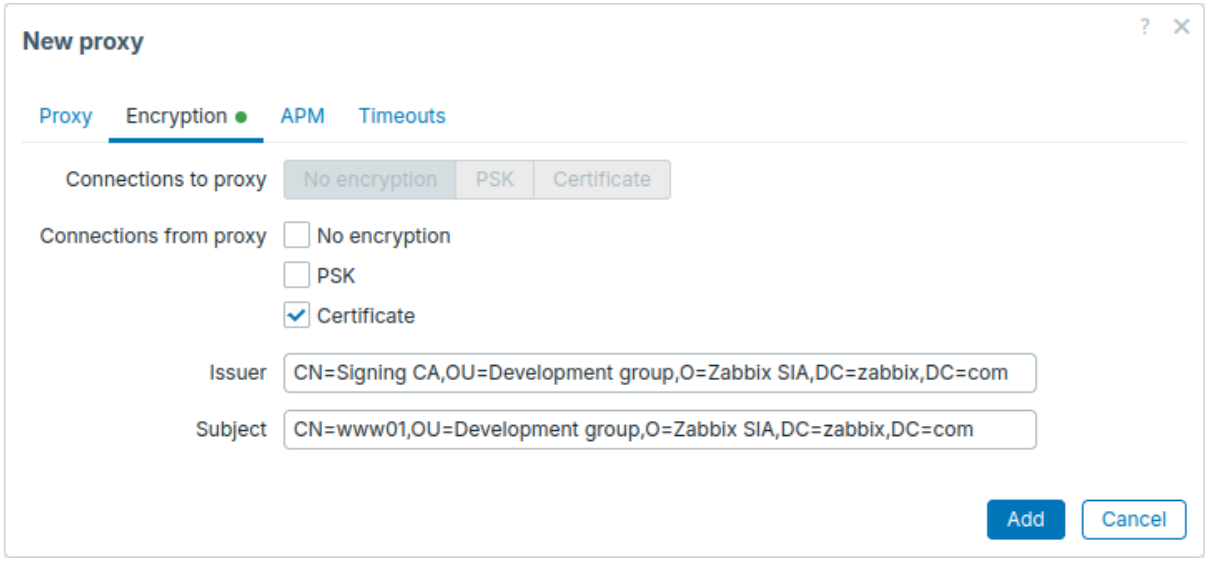
```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSServerCertIssuer=CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Zabbix server,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_proxy.crt
TLSKeyFile=/home/zabbix/zabbix_proxy.key
```

3. Skonfiguruj szyfrowanie dla tego proxy w frontend Zabbix:

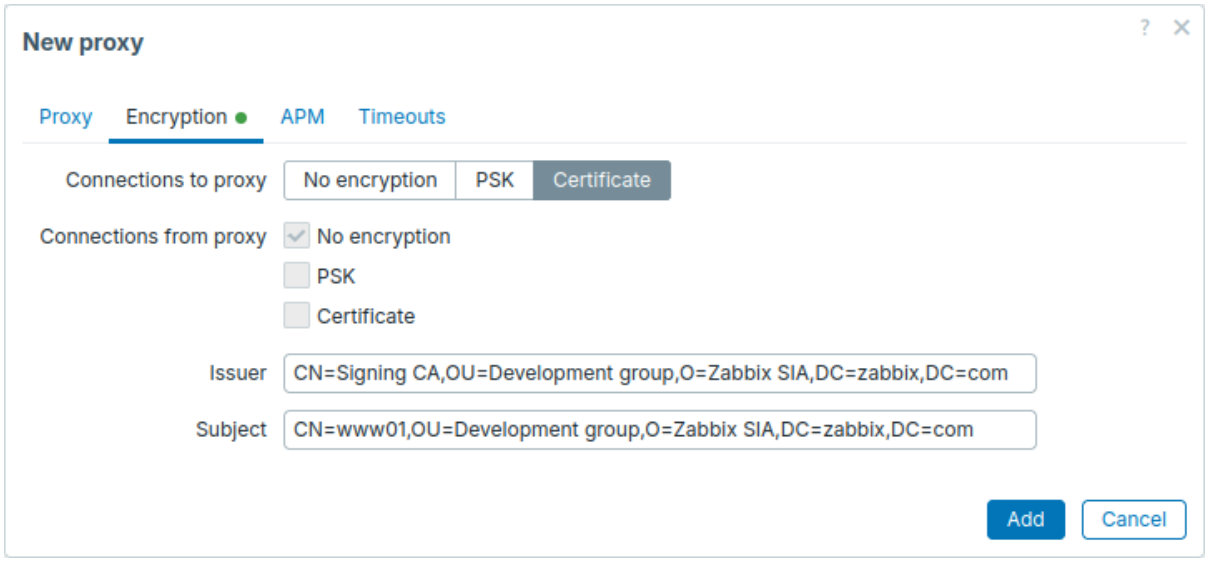
- Przejdź do: *Administracja* > *Proxy*.
- Wybierz proxy i kliknij kartę *Szyfrowanie*.

W poniższych przykładach pola *Wystawca* i *Podmiot* są wypełnione. Więcej informacji o tym, dlaczego i jak używać tych pól, znajdziesz w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.

Dla proxy aktywnego:



Dla proxy pasywnego:



Zabbix agent

1. Przygotuj pliki zawierające certyfikaty głównego urzędu certyfikacji (CA), certyfikat/łańcuch certyfikatów Zabbix agent oraz klucz prywatny, jak opisano w sekcji Zabbix server. Następnie odpowiednio edytuj parametry `TLSCAFile`, `TLSCertFile` i `TLSKeyFile` w pliku konfiguracyjnym agent Zabbix.

2. Edytuj dodatkowe parametry TLS w pliku konfiguracyjnym agent Zabbix:

- Dla agent aktywnego: `TLSConnect=cert`
- Dla agent pasywnego: `TLSAccept=cert`

Note:
Aby zwiększyć bezpieczeństwo agent, możesz ustawić parametry `TLSServerCertIssuer` i `TLSServerCertSubject`. Więcej informacji znajdziesz w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.

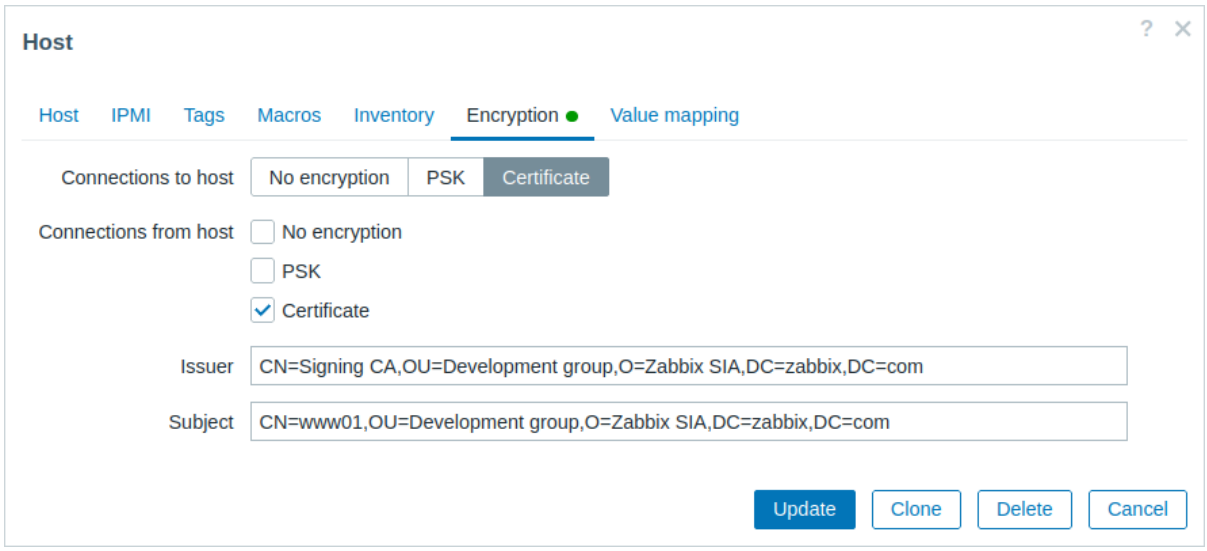
Parametry TLS w końcowym pliku konfiguracyjnym agent mogą wyglądać następująco. Należy pamiętać, że przykład zakłada, iż host jest monitorowany przez proxy, dlatego jest ono określone jako Subject:

```
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. Skonfiguruj szyfrowanie w frontend Zabbix dla host monitorowanego przez tego agent.

- Przejdź do: *Gromadzenie danych* > *Hosty*.
- Wybierz host i kliknij kartę *Szyfrowanie*.

W poniższym przykładzie wypełniono pola *Wystawca* i *Podmiot*. Więcej informacji o tym, dlaczego i jak używać tych pól, znajdziesz w sekcji Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu.



Usługa web Zabbix

1. Przygotuj pliki zawierające certyfikaty głównego urzędu certyfikacji (CA), certyfikat/łańcuch certyfikatów usługi web Zabbix oraz klucz prywatny, zgodnie z opisem w sekcji serwer Zabbix. Następnie odpowiednio edytuj parametry `TLSCAFile`, `TLSCertFile` i `TLSKeyFile` w pliku konfiguracyjnym usługi web Zabbix.

2. Edytuj dodatkowy parametr TLS w pliku konfiguracyjnym usługi web Zabbix: `TLSAccept=cert`

Parametry TLS w końcowym pliku konfiguracyjnym usługi web mogą wyglądać następująco:

```
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file.crt
TLSCertFile=/home/zabbix/zabbix_web_service.crt
TLSKeyFile=/home/zabbix/zabbix_web_service.key
```

3. Skonfiguruj serwer Zabbix tak, aby łączył się z usługą web Zabbix skonfigurowaną do obsługi TLS, edytując parametr `WebServiceURL` w pliku konfiguracyjnym serwera Zabbix:

```
WebServiceURL=https://example.com:443/report
```

Ograniczanie dozwolonego wystawcy i podmiotu certyfikatu Gdy dwa komponenty Zabbix (na przykład serwer i agent) ustanawiają połączenie TLS, wzajemnie weryfikują swoje certyfikaty. Jeśli certyfikat drugiej strony został podpisany przez zaufany urząd certyfikacji (z wstępnie skonfigurowanym certyfikatem najwyższego poziomu w `TLSCAFile`), jest ważny, nie wygasł i przechodzi pozostałe kontrole, komunikacja między komponentami może się odbywać. W tym najprostszym przypadku wystawca i podmiot certyfikatu nie są weryfikowane.

Stanowi to jednak zagrożenie: każda osoba posiadająca ważny certyfikat może podszyć się pod dowolną inną osobę lub komponent (na przykład certyfikat hosta może zostać użyty do podszycia się pod serwer). Może to być akceptowalne w małych środowiskach, w których certyfikaty są podpisywane przez dedykowany wewnętrzny urząd certyfikacji, a ryzyko podszycia się jest niewielkie, ale może nie być wystarczające w większych środowiskach lub środowiskach o wyższych wymaganiach bezpieczeństwa.

Jeśli certyfikaty najwyższego poziomu urzędu certyfikacji są wystawiane dla certyfikatów, które nie powinny być akceptowane przez Zabbix, lub jeśli chcesz zmniejszyć ryzyko podszycia się, możesz ograniczyć dozwolone certyfikaty, określając ich wystawcę i podmiot.

Na przykład w pliku konfiguracyjnym proxy Zabbix możesz określić:

```
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
```

Przy tych ustawieniach aktywny proxy nie będzie komunikować się z serwerem Zabbix, którego certyfikat ma innego wystawcę lub podmiot. Podobnie pasywny proxy nie będzie akceptować żądań od takiego serwera.

Zasady dopasowywania ciągów `Issuer` i `Subject`

Zasady dopasowywania ciągów `Issuer` i `Subject` są następujące:

- Ciągi `Issuer` i `Subject` są sprawdzane niezależnie. Oba są opcjonalne.
- Niespecyfikowany ciąg oznacza, że akceptowany jest dowolny ciąg.
- Ciągi są porównywane *tak jak są* i muszą być zgodne dokładnie.
- Obsługiwane są znaki UTF-8. Nie są jednak obsługiwane symbole wieloznaczne (`*`) ani wyrażenia regularne.
- Zaimplementowano następujące wymagania RFC 4514 - znaki wymagające poprzedzenia znakiem ucieczki (za pomocą odwrotnego ukośnika '\', U+005C):
  - w dowolnym miejscu ciągu: '"' (U+0022), '+' (U+002B), ',' (U+002C), ';' (U+003B), '<' (U+003C), '>' (U+003E), '\\' (U+005C);
  - na początku ciągu: spacja (' ', U+0020) lub znak numeru ('#', U+0023);
  - na końcu ciągu: spacja (' ', U+0020).
- Znaki null (U+0000) nie są obsługiwane. Jeśli zostanie napotkany znak null, dopasowanie zakończy się niepowodzeniem.
- Standardy RFC 4517 i RFC 4518 nie są obsługiwane.

Na przykład, jeśli ciągi organizacji (`O`) dla `Issuer` i `Subject` zawierają końcowe spacje, a ciąg jednostki organizacyjnej (`OU`) dla `Subject` zawiera podwójne cudzysłowy, znaki te muszą zostać poprzedzone znakiem ucieczki:

```
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
```

Kolejność i formatowanie pól

Zabbix stosuje zalecenia z RFC 4514, która określa "odwróconą" kolejność tych pól, zaczynając od pól najniższego poziomu (`CN`), przechodząc przez pola średniego poziomu (`OU`, `O`), a kończąc na polach najwyższego poziomu (`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
```

Natomiast OpenSSL domyślnie wyświetla ciągi `Issuer` i `Subject` w kolejności od najwyższego do najniższego poziomu. W poniższym przykładzie pola `Issuer` i `Subject` zaczynają się od pola najwyższego poziomu (`DC`), a kończą na polu najniższego poziomu (`CN`). Formatowanie ze spacjami i separatorami pól również różni się w zależności od użytych opcji, dlatego nie będzie zgodne z formatem wymaganym przez 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
```

Aby poprawnie sformatować ciągi *Issuer* i *Subject* dla Zabbix, wywołaj OpenSSL z następującymi opcjami:

```
$ 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
```

Wynik będzie wtedy w odwrotnej kolejności, rozdzielany przecinkami i będzie można go użyć w plikach konfiguracyjnych Zabbix oraz 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
```

Ograniczenia dotyczące używania rozszerzeń certyfikatów X.509 v3 Podczas implementowania certyfikatów X.509 v3 w Zabbix niektóre rozszerzenia mogą nie być w pełni obsługiwane lub mogą powodować niespójne zachowanie.

Rozszerzenie Subject Alternative Name

Zabbix nie obsługuje rozszerzenia *Subject Alternative Name*, które służy do określania alternatywnych nazw DNS, takich jak adresy IP lub adresy e-mail. Zabbix może zweryfikować wyłącznie wartość w polu *Subject* certyfikatu (zobacz Ograniczanie dozwolonego wystawcy i Subject certyfikatu). Jeśli certyfikaty zawierają pole `subjectAltName`, wynik weryfikacji certyfikatu może się różnić w zależności od konkretnych zestawów narzędzi kryptograficznych użytych do skompilowania komponentów Zabbix. W rezultacie Zabbix może zaakceptować lub odrzucić certyfikaty w zależności od tych kombinacji.

Rozszerzenie Extended Key Usage

Zabbix obsługuje rozszerzenie *Extended Key Usage*. Jeśli jednak jest ono używane, zazwyczaj wymagane jest określenie zarówno atrybutu *clientAuth* (dla uwierzytelniania klienta WWW TLS), jak i *serverAuth* (dla uwierzytelniania serwera WWW TLS). Na przykład:

- W przypadku kontroli pasywnych, gdzie agent Zabbix działa jako serwer TLS, atrybut *serverAuth* musi być uwzględniony w certyfikacie agenta.
- W przypadku kontroli aktywnych, gdzie agent działa jako klient TLS, atrybut *clientAuth* musi być uwzględniony w certyfikacie agenta.

Chociaż GnuTLS może zgłaszać ostrzeżenie dotyczące naruszeń użycia klucza, zazwyczaj pozwala na kontynuowanie komunikacji mimo tych ostrzeżeń.

Rozszerzenie Name Constraints

Obsługa rozszerzenia *Name Constraints* różni się w zależności od zestawu narzędzi kryptograficznych. Upewnij się, że wybrany zestaw narzędzi obsługuje to rozszerzenie. To rozszerzenie może uniemożliwić Zabbix wczytanie certyfikatów CA, jeśli ta sekcja jest oznaczona jako krytyczna, w zależności od używanego zestawu narzędzi.

Listy unieważnień certyfikatów (CRL) Jeśli certyfikat zostanie skompromitowany, urząd certyfikacji (CA) może go unieważnić, dodając certyfikat do listy unieważnień certyfikatów (CRL). CRL są zarządzane za pomocą plików konfiguracyjnych i można je określić przy użyciu parametru `TLSCRLFile` w plikach konfiguracyjnych serwer, proxy i agent. Na przykład:

```
TLSCRLFile=/home/zabbix/zabbix_crl_file.crt
```

W tym przypadku plik `zabbix_crl_file.crt` może zawierać CRL z wielu CA i może wyglądać następująco:

```
-----BEGIN X509 CRL-----
MIIB/DCB5QIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixkARkWA2Nv
...
treZeUPjb7LSmZ3K2hpbZN7SoOZcAoHQ3GWd9npuctg=
-----END X509 CRL-----
-----BEGIN X509 CRL-----
MIIB+TCB4gIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLGQBGRYDY29t
...
CAEebS2CND3ShBedZ8YSil59O6JvaDP61lR5lNs=
-----END X509 CRL-----
```

Plik CRL jest wczytywany tylko podczas uruchamiania Zabbixa. Aby zaktualizować CRL, uruchom ponownie Zabbixa.

> **Attention:**
> Jeśli komponenty Zabbixa są skompilowane z OpenSSL i używane są CRL, upewnij się, że każdy główny i pośredni CA w łańcuchach certyfikatów ma odpowiadającą mu CRL (nawet pustą) dołączoną w `TLSCRLFile`.

Rozwiązywanie problemów z certyfikatami OpenSSL używany z CRL, a dla niektórego CA w łańcuchu certyfikatów jego CRL nie jest uwzględniony w `TLSCRLFile`

W dzienniku serwera TLS w przypadku peera *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"
```

W dzienniku serwera TLS w przypadku peera *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 wygasł lub wygasa podczas działania serwera

OpenSSL, w logu serwera:

- przed wygaśnięciem:

```
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"
```

- po wygaśnięciu:

```
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"
```

Chodzi o to, że przy prawidłowym CRL odwołany certyfikat jest zgłaszany jako ”certificate revoked”. Gdy CRL wygasa, komunikat o błędzie zmienia się na ”certificate expired”, co jest dość mylące.

GnuTLS, w logu serwera:

- przed i po wygaśnięciu tak samo:

```
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.
```

Samopodpisany certyfikat, nieznany CA

OpenSSL, w logu:

```
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"'
```

Zaobserwowano to, gdy certyfikat serwera przez pomyłkę miał ten sam ciąg Issuer i Subject, mimo że został podpisany przez CA. Issuer i Subject są równe w głównym certyfikacie CA, ale nie mogą być równe w certyfikacie serwera. (To samo dotyczy certyfikatów proxy i agent.)

Aby sprawdzić, czy certyfikat zawiera te same wpisy Issuer i Subject, uruchom:

```
openssl x509 -in <yourcertificate.crt> -noout -text
```

Dopuszczalne jest, aby certyfikat główny (najwyższego poziomu) miał identyczne wartości Issuer i Subject.

Szyfrowanie przy użyciu kluczy współdzielonych z wyprzedzeniem

Omówienie Każdy klucz wstępnie współdzielony (PSK) w Zabbixie jest w rzeczywistości parą:

- niezawierającego sekretu ciągu identyfikatora PSK,
- tajnego ciągu wartości PSK.

Ciąg identyfikatora PSK jest niepustym ciągiem znaków UTF-8. Na przykład: „PSK ID 001 Zabbix agentd”. Jest to unikalna nazwa, za pomocą której komponenty Zabbixa odwołują się do tego konkretnego PSK. Nie umieszczaj poufnych informacji w ciągu identyfikatora PSK — jest on przesyłany przez sieć w postaci niezaszyfrowanej.

Wartość PSK to trudny do odgadnięcia ciąg cyfr szesnastkowych, na przykład e560cb0d918d26d31b4f642181f5f570ad89a390931102e53

Warning:
Nigdy nie używaj przykładowych wartości PSK podanych w dokumentacji w środowiskach produkcyjnych. Zamiast tego wygeneruj unikalny, kryptograficznie bezpieczny PSK.

Limity rozmiaru W Zabbix obowiązują limity rozmiaru tożsamości PSK i wartości PSK. W niektórych przypadkach biblioteka kryptograficzna może mieć niższy limit:

Komponent	Maksymalny rozmiar tożsamości PSK	Minimalny rozmiar wartości PSK	Maksymalny rozmiar wartości PSK
Zabbix	128 znaków UTF-8	128 bitów (PSK o rozmiarze 16 bajtów, wprowadzany jako 32 cyfry szesnastkowe)	2048 bitów (PSK o rozmiarze 256 bajtów, wprowadzany jako 512 cyfr szesnastkowych)
GnuTLS	128 bajtów (może zawierać znaki UTF-8)	-	2048 bitów (PSK o rozmiarze 256 bajtów, wprowadzany jako 512 cyfr szesnastkowych)
OpenSSL 1.0.x, 1.1.0	127 bajtów (może zawierać znaki UTF-8)	-	2048 bitów (PSK o rozmiarze 256 bajtów, wprowadzany jako 512 cyfr szesnastkowych)
OpenSSL 1.1.1	127 bajtów (może zawierać znaki UTF-8)	-	512 bitów (PSK o rozmiarze 64 bajtów, wprowadzany jako 128 cyfr szesnastkowych)
OpenSSL 1.1.1a i nowsze	127 bajtów (może zawierać znaki UTF-8)	-	2048 bitów (PSK o rozmiarze 256 bajtów, wprowadzany jako 512 cyfr szesnastkowych)

Attention:
frontend Zabbix umożliwia skonfigurowanie tożsamości PSK o długości do 128 znaków oraz PSK o długości 2048 bitów, niezależnie od używanych bibliotek kryptograficznych.
 Jeśli niektóre komponenty Zabbix obsługują niższe limity, użytkownik jest odpowiedzialny za skonfigurowanie tożsamości i wartości PSK o długości dozwolonej dla tych komponentów.
 Przekroczenie limitów długości powoduje błędy komunikacji między komponentami Zabbix.

Zanim serwer Zabbix połączy się z agentem przy użyciu PSK, serwer wyszukuje w bazie danych (a dokładniej w pamięci podręcznej konfiguracji) tożsamość PSK i wartość PSK skonfigurowane dla tego agenta. Po otrzymaniu połączenia agent używa tożsamości PSK i wartości PSK z jego pliku konfiguracyjnego. Jeśli obie strony mają ten sam ciąg tożsamości PSK i tę samą wartość PSK, połączenie może się powieść.

Attention:
Każda tożsamość PSK musi być powiązana tylko z jedną wartością. Użytkownik jest odpowiedzialny za zapewnienie, że nie istnieją dwie wartości PSK z tym samym ciągiem tożsamości, ale z różnymi wartościami. Niezastosowanie się do tego zalecenia może prowadzić do nieprzewidywalnych błędów lub zakłóceń komunikacji między komponentami Zabbix używającymi PSK z tym ciągiem tożsamości PSK.

Generowanie PSK Na przykład klucz PSK o długości 256 bitów (32 bajtów) można wygenerować za pomocą następujących poleceń:

- za pomocą *OpenSSL*:

```
$ openssl rand -hex 32
af8ced32dfe8714e548694e2d29e1a14ba6fa13f216cb35c19d0feb1084b0429
```

- za pomocą *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
```

Należy pamiętać, że polecenie „psktool” powyżej generuje plik bazy danych zawierający tożsamość PSK i powiązany z nią klucz PSK. Zabbix oczekuje w pliku PSK wyłącznie klucza PSK, dlatego z pliku należy usunąć ciąg tożsamości oraz dwukropek („:”).

Konfigurowanie PSK na potrzeby komunikacji serwer-agent (przykład) Na hoście agenta zapisz wygenerowaną wcześniej wartość PSK w pliku, na przykład `/home/zabbix/zabbix_agentd.psk`. Plik musi zawierać wartość PSK w pierwszym wierszu.

Ustaw prawa dostępu do pliku PSK — musi on być dostępny do odczytu wyłącznie dla użytkownika Zabbix.

Edytuj parametry TLS w pliku konfiguracyjnym agenta Zabbix (`zabbix_agentd.conf`). Na przykład ustaw:

```
TLSConnect=psk
TLSAccept=psk
TLSPSKFile=/home/zabbix/zabbix_agentd.psk
TLSPSKIdentity=PSK 001
```

agent będzie łączyć się z serwerem (aktywne sprawdzanie) i akceptować z serwera oraz `zabbix_get` wyłącznie połączenia korzystające z PSK. Tożsamość PSK będzie mieć wartość „PSK 001”.

Uruchom ponownie agenta. Teraz możesz przetestować połączenie za pomocą `zabbix_get`, na przykład:

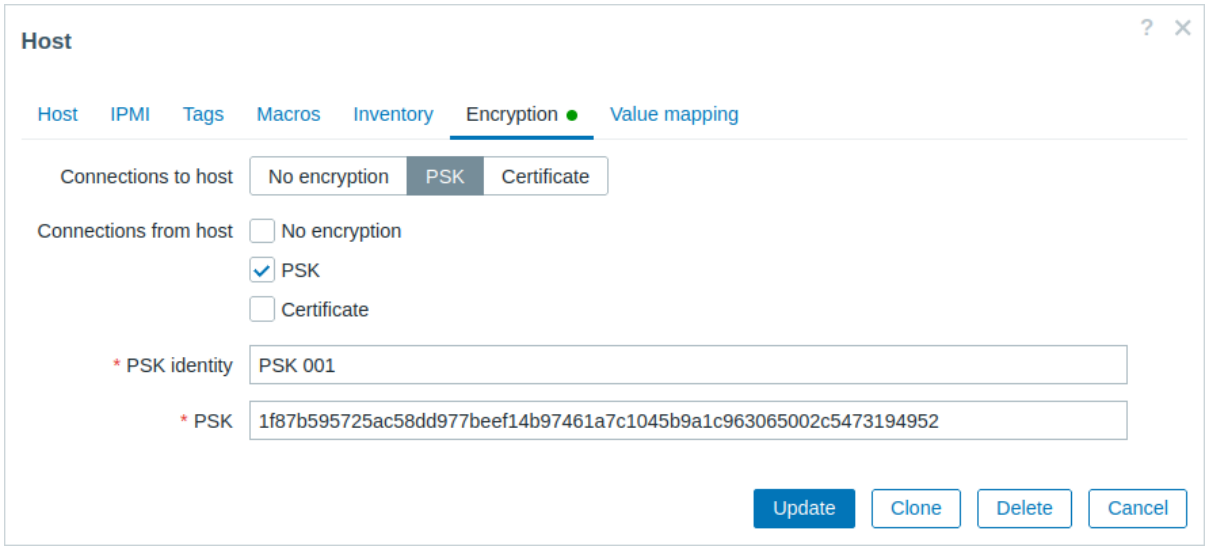
```
zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk --tls-psk-identity="PSK 001" --tl
```

(Aby zminimalizować przestój, zobacz, jak zmienić typ połączenia w sekcji Zarządzanie szyfrowaniem połączeń).

Skonfiguruj szyfrowanie PSK dla tego agenta w frontend Zabbix:

1. Przejdź do: *Zbieranie danych* > *Hosty*.
2. Wybierz host i kliknij kartę *Szyfrowanie*.
3. Skonfiguruj szczegóły szyfrowania PSK.

Przykład:



Wszystkie wymagane pola są oznaczone czerwoną gwiazdką.

Gdy pamięć podręczna konfiguracji zostanie zsynchronizowana z bazą danych, nowe połączenia będą korzystać z PSK. Sprawdź pliki dziennika serwera i agenta pod kątem komunikatów o błędach.

Konfigurowanie PSK na potrzeby komunikacji serwer - proxy aktywne (przykład) Na proxy zapisz wartość PSK w pliku, na przykład `/home/zabbix/zabbix_proxy.psk`. Plik musi zawierać PSK w pierwszym wierszu tekstu, na przykład:

```
e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9
```

Ustaw prawa dostępu do pliku PSK - musi on być dostępny do odczytu wyłącznie dla użytkownika Zabbix.

Edytuj parametry TLS w pliku konfiguracyjnym proxy Zabbix (`zabbix_proxy.conf`). Na przykład ustaw:

```
TLSConnect=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

proxy połączy się z serwerem przy użyciu PSK. Tożsamość PSK będzie mieć wartość „PSK 002”.

(Aby zminimalizować przestój, zobacz, jak zmienić typ połączenia w sekcji Zarządzanie szyfrowaniem połączeń).

Skonfiguruj PSK dla tego proxy w frontend Zabbix. Przejdź do *Administracja* > *Proxy*, wybierz proxy i przejdź do karty *Szyfrowanie*. W sekcji „Połączenia z proxy" zaznacz PSK. W polu „Tożsamość PSK" wklej „PSK 002", a w polu „PSK" wklej „e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9". Kliknij „Aktualizuj".

Uruchom ponownie proxy. Zacznie ono używać szyfrowanych połączeń z serwerem opartych na PSK. Sprawdź pliki dziennika serwera i proxy pod kątem komunikatów o błędach.

W przypadku proxy pasywnego procedura jest bardzo podobna. Jedyna różnica polega na ustawieniu TLSAccept=psk w pliku konfiguracyjnym proxy Zabbix oraz ustawieniu opcji *Połączenia z proxy* w frontend Zabbix na PSK.

Rozwiązywanie problemów z PSK PSK zawiera nieparzystą liczbę cyfr szesnastkowych

Proxy lub agent nie uruchamia się, komunikat w logu proxy lub agenta:

```
nieprawidłowy PSK w pliku "/home/zabbix/zabbix_proxy.psk"
```

Ciąg tożsamości PSK dłuższy niż 128 bajtów jest przekazywany do GnuTLS

W logu po stronie klienta TLS:

```
gnutls_handshake() failed: -110 The TLS connection was non-properly terminated.
```

W logu po stronie serwera TLS:

```
gnutls_handshake() failed: -90 The SRP username supplied is illegal.
```

Zbyt długa wartość PSK użyta z OpenSSL 1.1.1

W logu strony łączącej:

```
...Biblioteka OpenSSL (wersja OpenSSL 1.1.1  11 Sep 2018) zainicjalizowana
...
...W zbx_tls_connect(): psk_identity:"PSK 1"
...zbx_psk_client_cb() zażądał tożsamości PSK "PSK 1"
...Koniec zbx_tls_connect():FAIL błąd:'SSL_connect() set result code to SSL_ERROR_SSL: file ssl\statem\ext
```

W logu strony przyjmującej:

```
...Wiadomość od 123.123.123.123 nie zawiera nagłówka. Wiadomość zignorowana.
```

Ten problem zwykle występuje podczas aktualizacji OpenSSL z wersji 1.0.x lub 1.1.0 do 1.1.1, jeśli wartość PSK jest dłuższa niż 512 bitów (PSK 64-bajtowy, wprowadzony jako 128 cyfr szesnastkowych).

Zobacz też: Ograniczenia rozmiaru wartości

Bezpieczne połączenie z bazą danych

Omówienie

Ta sekcja zawiera kroki konfiguracji Zabbix oraz przykłady konfiguracji dla bezpiecznych połączeń TLS między:

Database	Zabbix components
MySQL	Zabbix frontend, Zabbix serwer, Zabbix proxy
PostgreSQL	Zabbix frontend, Zabbix serwer, Zabbix proxy

Aby skonfigurować szyfrowanie połączeń w obrębie DBMS, zapoznaj się ze szczegółami w oficjalnej dokumentacji producenta:

- MySQL: serwery baz danych replikacji źródłowej i repliki.
- MySQL: serwery baz danych replikacji grupowej itp.
- PostgreSQL opcje szyfrowania.

Wszystkie przykłady opierają się na wydaniach GA MySQL CE (8.0) i PgSQL (13) dostępnych w oficjalnych repozytoriach przy użyciu CentOS 8.

Wymagania

Do skonfigurowania szyfrowania wymagane jest:

- System operacyjny wspierany przez producenta, z OpenSSL >=1.1.X lub alternatywą.

Note:
Zaleca się unikanie systemów operacyjnych w stanie end-of-life, szczególnie w przypadku nowych instalacji

- Silnik bazy danych (RDBMS) zainstalowany i utrzymywany z oficjalnego repozytorium udostępnianego przez producenta. Systemy operacyjne często są dostarczane z przestarzałymi wersjami oprogramowania bazodanowego, dla których obsługa szyfrowania nie jest zaimplementowana, na przykład systemy oparte na RHEL 7 oraz PostgreSQL 9.2, MariaDB 5.5 bez obsługi szyfrowania.

Terminologia

Ustawienie tej opcji wymusza użycie połączenia TLS z bazą danych z serwera/proxy Zabbix oraz frontend do bazy danych:

- `required` - połączenie z użyciem TLS jako trybu transportu bez sprawdzania tożsamości
- `verify_ca` - połączenie z użyciem TLS i weryfikacją certyfikatu
- `verify_full` - połączenie z użyciem TLS, weryfikacją certyfikatu oraz sprawdzeniem, czy tożsamość bazy danych (CN) określona przez DBHost jest zgodna z jej certyfikatem

Konfiguracja Zabbix

Frontend do bazy danych

Bezpieczne połączenie z bazą danych można skonfigurować podczas instalacji frontendu:

- Zaznacz pole wyboru *Szyfrowanie TLS bazy danych* w kroku Konfiguruj połączenie z bazą danych, aby włączyć szyfrowanie transmisji.
- Zaznacz pole wyboru *Weryfikuj certyfikat bazy danych*, które pojawia się po zaznaczeniu pola *Szyfrowanie TLS*, aby włączyć szyfrowanie za pomocą certyfikatów.

Note:
W przypadku MySQL pole wyboru *Szyfrowanie TLS bazy danych* jest wyłączone, jeśli *Host bazy danych* jest ustawiony na localhost, ponieważ połączenie korzystające z pliku gniazda (w systemie Unix) lub pamięci współdzielonej (w systemie Windows) nie może być szyfrowane.
 W przypadku PostgreSQL pole wyboru *Szyfrowanie TLS* jest wyłączone, jeśli wartość pola *Host bazy danych* zaczyna się od ukośnika lub pole jest puste.

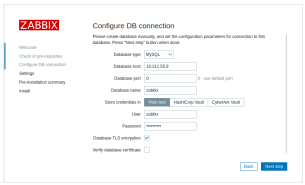
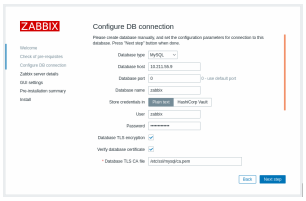
W trybie szyfrowania TLS za pomocą certyfikatów (jeśli oba pola wyboru są zaznaczone) dostępne są następujące parametry:

Parametr	Opis
Plik CA TLS bazy danych	Określ pełną ścieżkę do prawidłowego pliku urzędu certyfikacji (CA) TLS.
Plik klucza TLS bazy danych	Określ pełną ścieżkę do prawidłowego pliku klucza TLS.
Plik certyfikatu TLS bazy danych	Określ pełną ścieżkę do prawidłowego pliku certyfikatu TLS.
Weryfikacja hosta bazy danych	Zaznacz to pole wyboru, aby aktywować weryfikację hosta. Wyłączone dla MySQL, ponieważ biblioteka PHP MySQL nie pozwala pominąć etapu walidacji certyfikatu peera.
Lista szyfrów TLS bazy danych	Określ niestandardową listę prawidłowych szyfrów. Format listy szyfrów musi być zgodny ze standardem OpenSSL. Dostępne tylko dla MySQL.

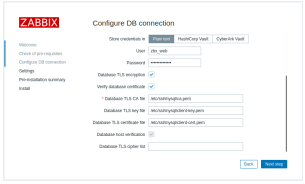
Attention:
Parametry TLS muszą wskazywać prawidłowe pliki. Jeśli wskazują nieistniejące lub nieprawidłowe pliki, spowoduje to błąd autoryzacji.
 Jeśli pliki certyfikatów są zapisywalne, frontend generuje ostrzeżenie w raporcie Informacje o systemie: „Pliki certyfikatów TLS muszą być tylko do odczytu”. (Wyświetlane tylko wtedy, gdy użytkownik PHP jest właścicielem certyfikatu).
Certyfikaty chronione hasłami nie są obsługiwane.

Przypadki użycia

frontend Zabbix używa interfejsu GUI do definiowania dostępnych opcji: `required`, `verify_ca`, `verify_full`. Określ wymagane opcje w kroku kreatora instalacji *Konfigurowanie połączeń z bazą danych*. Opcje te są mapowane na plik konfiguracyjny frontend (`zabbix.conf.php`) w następujący sposób:

Ustawienia GUI	Plik konfiguracyjny	Opis	Wynik
	... // Used for TLS connection. $DB['ENCRYPTION'] = true; $DB['KEY_FILE'] = ''; $DB['CERT_FILE'] = ''; $DB['CA_FILE'] = ''; $DB['VERIFY_HOST'] = false; $DB['CIPHER_LIST'] = ''; ...	1. Zaznacz *Szyfrowanie TLS bazy danych* 2. Pozostaw opcję *Weryfikuj certyfikat bazy danych* niezaznaczoną	Włącz tryb `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. Zaznacz opcje *Szyfrowanie TLS bazy danych* i *Weryfikuj certyfikat bazy danych* 2. Określ ścieżkę do *pliku CA TLS bazy danych*	Włącz tryb `verify_ca`.

Ustawienia GUI	Plik konfiguracyjny	Opis	Wynik

	...
	// 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>';`
	...
	
	Lub:
	
	...
	// 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. Zaznacz opcje *Szyfrowanie TLS bazy danych* i *Weryfikuj certyfikat bazy danych*
		2. Określ ścieżkę do *pliku klucza TLS bazy danych*
		3. Określ ścieżkę do *pliku CA TLS bazy danych*
		4. Określ ścieżkę do *pliku certyfikatu TLS bazy danych*
		5. Określ *listę szyfrów TLS bazy danych* (opcjonalnie)
			Włącz tryb `verify_full` dla MySQL.

Ustawienia GUI	Plik konfiguracyjny	Opis	Wynik

	...
	`$DB['ENCRYPTION'] = true;`
	`$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. Zaznacz opcje *Szyfrowanie TLS bazy danych* i *Weryfikuj certyfikat bazy danych*
		2. Określ ścieżkę do *pliku klucza TLS bazy danych*
		3. Określ ścieżkę do *pliku CA TLS bazy danych*
		4. Określ ścieżkę do *pliku certyfikatu TLS bazy danych*
		5. Zaznacz opcję *Weryfikacja hosta bazy danych*
			Włącz tryb `verify_full` dla PostgreSQL.

Zobacz także: Przykłady konfiguracji szyfrowania dla MySQL, Przykłady konfiguracji szyfrowania dla PostgreSQL.

Konfiguracja serwera/proxy Zabbix

Bezpieczne połączenia z bazą danych można skonfigurować za pomocą odpowiednich parametrów w pliku konfiguracyjnym serwera Zabbix i/lub proxy.

Konfiguracja	Wynik
Brak	Połączenie z bazą danych bez szyfrowania.
Ustaw `DBTLSConnect=required`	Serwer/proxy nawiązuje połączenie TLS z bazą danych. Połączenie nieszyfrowane jest niedozwolone.
1. Ustaw `DBTLSConnect=verify_ca` 2. Ustaw `DBTLSCAFile` - określ plik urzędu certyfikacji TLS	Serwer/proxy nawiązuje połączenie TLS z bazą danych po zweryfikowaniu certyfikatu bazy danych.
1. Ustaw `DBTLSConnect=verify_full` 2. Ustaw `DBTLSCAFile` - określ plik urzędu certyfikacji TLS	Serwer/proxy nawiązuje połączenie TLS z bazą danych po zweryfikowaniu certyfikatu bazy danych i tożsamości hosta bazy danych.
1. Ustaw `DBTLSCAFile` - określ plik urzędu certyfikacji TLS 2. Ustaw `DBTLSCertFile` - określ plik certyfikatu klucza publicznego klienta 3. Ustaw `DBTLSKeyFile` - określ plik klucza prywatnego klienta	Serwer/proxy przedstawia certyfikat klienta podczas łączenia się z bazą danych.
Ustaw `DBTLSCipher` - listę szyfrów szyfrowania dozwolonych przez klienta dla połączeń korzystających z protokołów TLS do wersji TLS 1.2 włącznie lub `DBTLSCipher13` - listę szyfrów szyfrowania dozwolonych przez klienta dla połączeń korzystających z protokołu TLS 1.3	(MySQL) Połączenie TLS jest nawiązywane przy użyciu szyfru z podanej listy. (PostgreSQL) Ustawienie tej opcji zostanie uznane za błąd.

Konfiguracja szyfrowania MySQL

Omówienie

Ta sekcja zawiera kilka przykładów konfiguracji szyfrowania dla CentOS 8.2 i MySQL 8.4.0 i może służyć jako przewodnik szybkiego startu do szyfrowania połączenia z bazą danych.

Attention:
Jeśli host MySQL jest ustawiony na localhost, opcje szyfrowania nie będą dostępne. W takim przypadku połączenie między frontendem Zabbix a bazą danych korzysta z pliku gniazda (na Unix) lub pamięci współdzielonej (na Windows) i nie może być szyfrowane.

Note:
Lista kombinacji szyfrowania nie ogranicza się do tych wymienionych na tej stronie. Dostępnych jest znacznie więcej kombinacji.

Wymagania wstępne

Zainstaluj bazę danych MySQL z oficjalnego repozytorium.

Szczegółowe informacje o korzystaniu z repozytorium MySQL znajdziesz w dokumentacji MySQL.

serwer MySQL jest gotowy do akceptowania bezpiecznych połączeń przy użyciu certyfikatu samopodpisanego.

Aby sprawdzić, którzy użytkownicy korzystają z szyfrowanego połączenia, uruchom następujące zapytanie (Performance Schema powinno być włączone):

```
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;
```

Szyfrowanie tylko transportu

Konfiguracja MySQL

Nowoczesne wersje bazy danych są gotowe do użycia od razu po instalacji w przypadku `required` trybu szyfrowania. Certyfikat po stronie serwera zostanie utworzony po wstępnej konfiguracji i uruchomieniu.

Utwórz użytkowników i role dla głównych komponentów (zastąp <strong_password> silnym, unikatowym hasłem):

```
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'@'%';
```

Należy pamiętać, że protokół X.509 nie jest używany do sprawdzania tożsamości, ale użytkownik jest skonfigurowany tak, aby korzystać wyłącznie z szyfrowanych połączeń. Więcej informacji na temat konfigurowania użytkowników można znaleźć w dokumentacji MySQL.

Uruchom poniższe polecenie, aby sprawdzić połączenie (połączenie za pomocą gniazda nie może być używane do testowania bezpiecznych połączeń):

```
mysql -u zbx_srv -p -h 10.211.55.9 --ssl-mode=REQUIRED
```

Sprawdź bieżący stan i dostępne zestawy szyfrów:

```
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

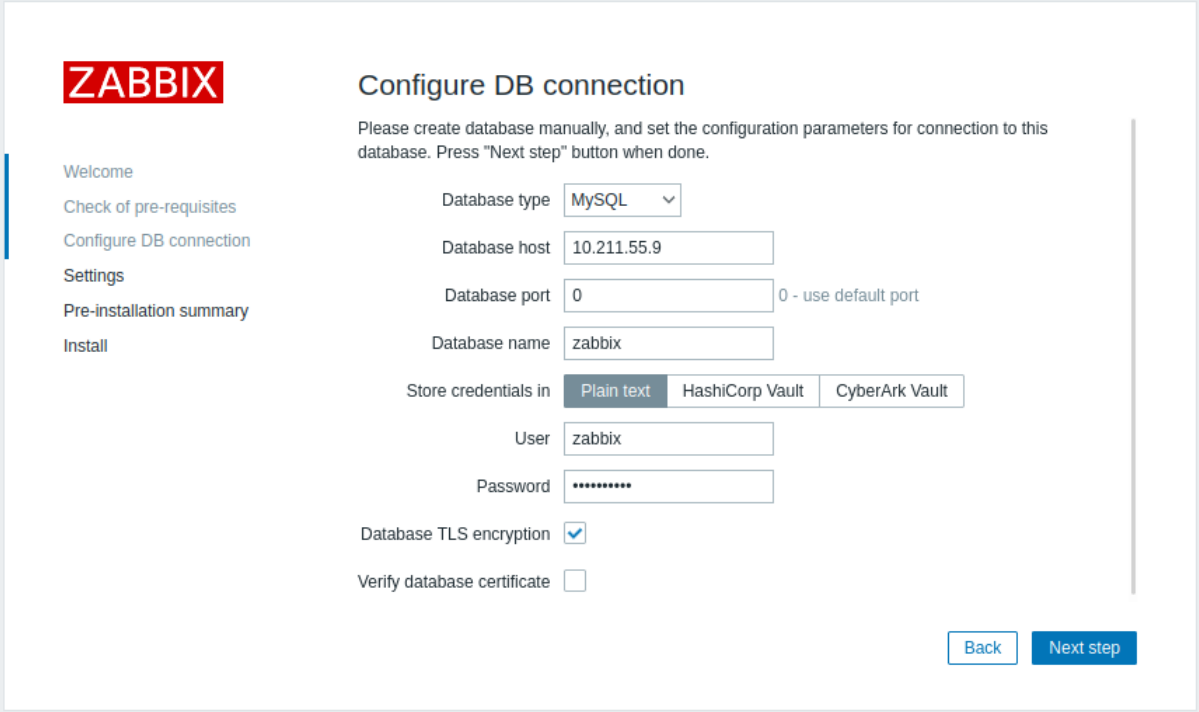

mysql> SHOW SESSION STATUS LIKE 'Ssl_cipher_list'\G;
*************************** 1. row ***************************
Variable_name: Ssl_cipher_list
Value: TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_SHA256:E
1 row in set (0.00 sec)

ERROR:
No query specified
```

Frontend

Aby włączyć szyfrowanie wyłącznie transportowe dla połączeń między frontendem Zabbixa a bazą danych:

- Zaznacz *Database TLS encryption*
- Pozostaw niezaznaczone *Verify database certificate*



Serwer

Aby włączyć szyfrowanie wyłącznie transportowe dla połączeń między serwerem a bazą danych, edytuj plik konfiguracyjny serwera:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Szyfrowanie z weryfikacją urzędu certyfikacji

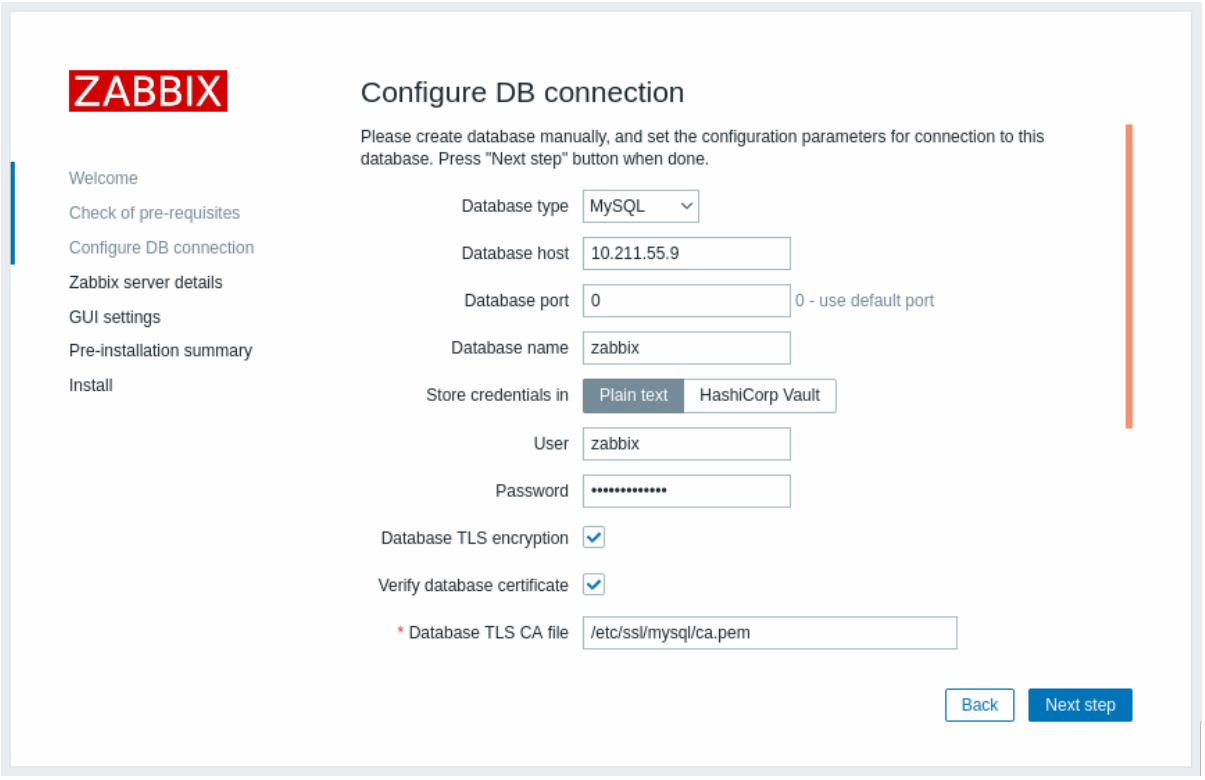
Skopiuj wymagany certyfikat CA MySQL na serwer frontend Zabbixa i nadaj odpowiednie uprawnienia, aby serwer WWW mógł odczytać ten plik.

Note:
Ten tryb nie działa w RHEL 7 z powodu starszych bibliotek MySQL.

Frontend

Aby włączyć szyfrowanie z weryfikacją certyfikatu dla połączeń między frontend a bazą danych:

- Zaznacz *Szyfrowanie TLS bazy danych* oraz *Weryfikuj certyfikat bazy danych*
- Określ ścieżkę do pliku CA TLS bazy danych



Alternatywnie można to ustawić w pliku konfiguracyjnym 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'] = '';
...
```

Rozwiąż problem, używając narzędzia wiersza poleceń do sprawdzenia, czy połączenie jest możliwe dla wymaganego użytkownika:

```
    mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=REQUIRED --ssl-ca=/var/lib/mysql/ca.pem
```

Serwer

Aby włączyć szyfrowanie z weryfikacją certyfikatu dla połączeń między serwerem Zabbix a bazą danych, edytuj plik konfiguracyjny serwera:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/mysql/ca.pem
...
```

Szyfrowanie z pełną weryfikacją

Konfiguracja MySQL

Ustaw opcję konfiguracji serwera MySQL CE (*/etc/my.cnf.d/server-tls.cnf*) na:

```
[mysqld]
...
#### w tym przykładzie klucze znajdują się w katalogu datadir MySQL CE
ssl_ca=ca.pem
```

```
ssl_cert=server-cert.pem
ssl_key=server-key.pem

require_secure_transport=ON
tls_version=TLSv1.3
...
```

Klucze dla serwera i klienta MySQL CE (frontend Zabbix) powinny zostać utworzone ręcznie zgodnie z dokumentacją MySQL CE: Creating SSL and RSA certificates and keys using MySQL lub Creating SSL certificates and keys using openssl

Attention:
Certyfikat serwera MySQL powinien zawierać pole Common Name ustawione na nazwę FQDN, ponieważ frontend Zabbix będzie używać nazwy DNS do komunikacji z bazą danych lub adresu IP hosta bazy danych.

Utwórz użytkownika 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;
```

Sprawdź, czy można zalogować się tym użytkownikiem:

```
mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=VERIFY_IDENTITY --ssl-ca=/var/lib/mysql/ca.pem --ssl-cert=/v
```

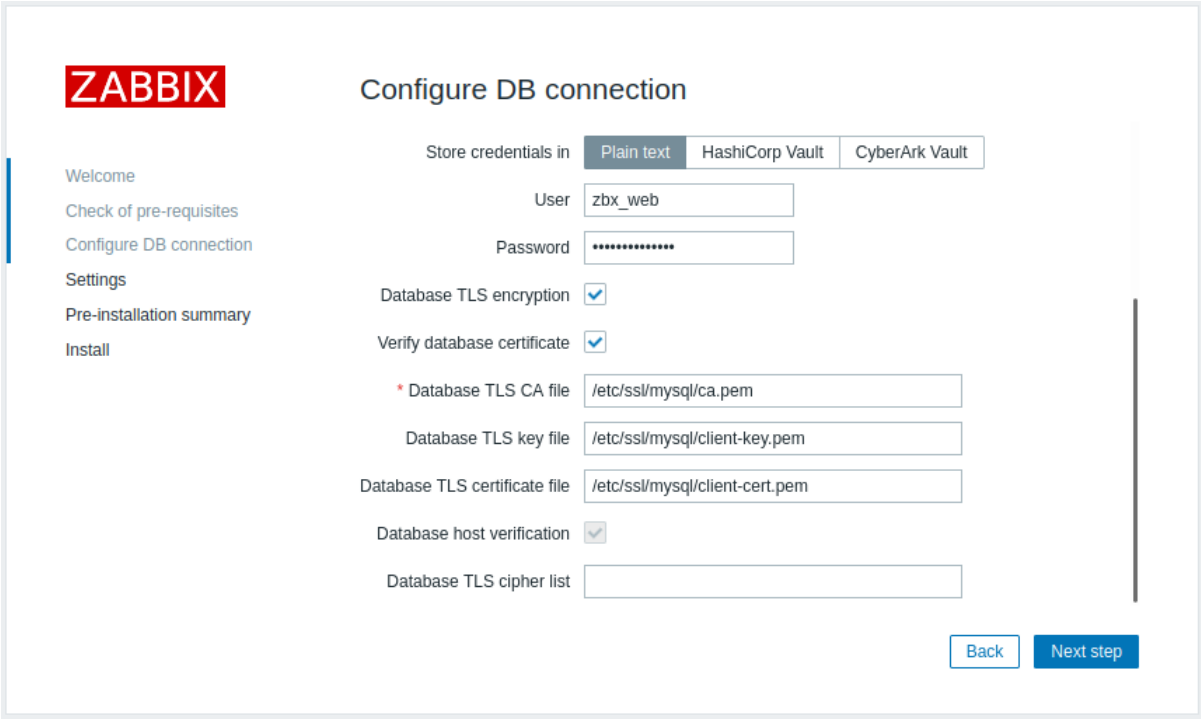
Frontend

Aby włączyć szyfrowanie z pełną weryfikacją połączeń między frontendem Zabbix a bazą danych:

- Zaznacz *Szyfrowanie TLS bazy danych* oraz *Weryfikuj certyfikat bazy danych*
- Określ ścieżkę do *pliku klucza TLS bazy danych*
- Określ ścieżkę do *pliku CA TLS bazy danych*
- Określ ścieżkę do *pliku certyfikatu TLS bazy danych*

Zwróć uwagę, że opcja *Weryfikacja hosta bazy danych* jest zaznaczona i wyszarzona - tego kroku nie można pominąć w przypadku MySQL.

Warning:
Jeśli pole *Lista szyfrów TLS bazy danych* pozostanie puste, zostaną włączone wspólne szyfry dozwolone zarówno przez frontend (klienta), jak i serwer. Alternatywnie szyfry można ustawić jawnie, zgodnie z wymaganiami dotyczącymi konfiguracji szyfrów.



Alternatywnie można to ustawić w pliku konfiguracyjnym frontendu (`zabbix.conf.php`):

```
...
// Używane do połączenia TLS ze ściśle określoną listą szyfrów.
$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
...
// lub
...
// Używane do połączenia TLS bez zdefiniowanej listy szyfrów - wybieranej przez serwer 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'] = '';
...
```

Serwer

Aby włączyć szyfrowanie z pełną weryfikacją połączeń między serwerem Zabbix a bazą danych, edytuj plik konfiguracyjny serwera:

```
...
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
...
```

Konfiguracja szyfrowania PostgreSQL

Przegląd

Ta sekcja zawiera kilka przykładów konfiguracji szyfrowania dla CentOS 8.2 i PostgreSQL 13.

> **Note:**
> Połączenie między frontend Zabbix a PostgreSQL nie może być szyfrowane (parametry w GUI są wyłączone), jeśli wartość pola *Database host* zaczyna się od ukośnika lub pole jest puste.

Wymagania wstępne

Zainstaluj bazę danych PostgreSQL, korzystając z oficjalnego repozytorium.

PostgreSQL nie jest domyślnie skonfigurowany do akceptowania połączeń TLS. Postępuj zgodnie z instrukcjami z dokumentacji PostgreSQL dotyczącymi przygotowania certyfikatów z plikiem postgresql.conf oraz kontroli dostępu użytkowników za pomocą `ph_hba.conf`.

Domyślnie gniazdo PostgreSQL jest powiązane z localhost, aby dla zdalnych połączeń sieciowych umożliwić nasłuchiwanie na rzeczywistym interfejsie sieciowym.

Ustawienia PostgreSQL dla wszystkich trybów mogą wyglądać następująco:

`/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'
...
```

Dla kontroli dostępu dostosuj `/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

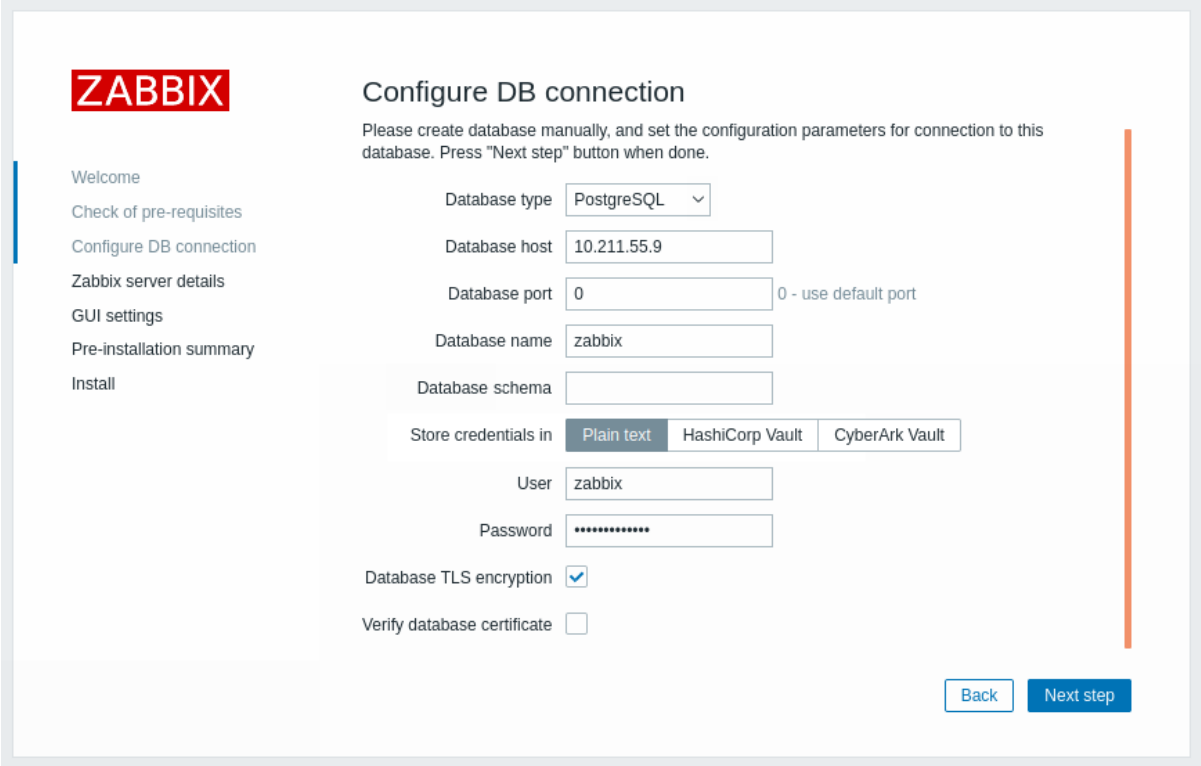
### verify full
hostssl all all 0.0.0.0/0 md5 clientcert=verify-full
...
```

Szyfrowanie tylko transportu

Frontend

Aby włączyć szyfrowanie tylko transportu dla połączeń między frontendem Zabbixa a bazą danych:

- Zaznacz *Database TLS encryption*
- Pozostaw *Verify database certificate* niezaznaczone



Serwer

Aby włączyć szyfrowanie wyłącznie transportowe dla połączeń między serwerem a bazą danych, edytuj plik konfiguracyjny serwera:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

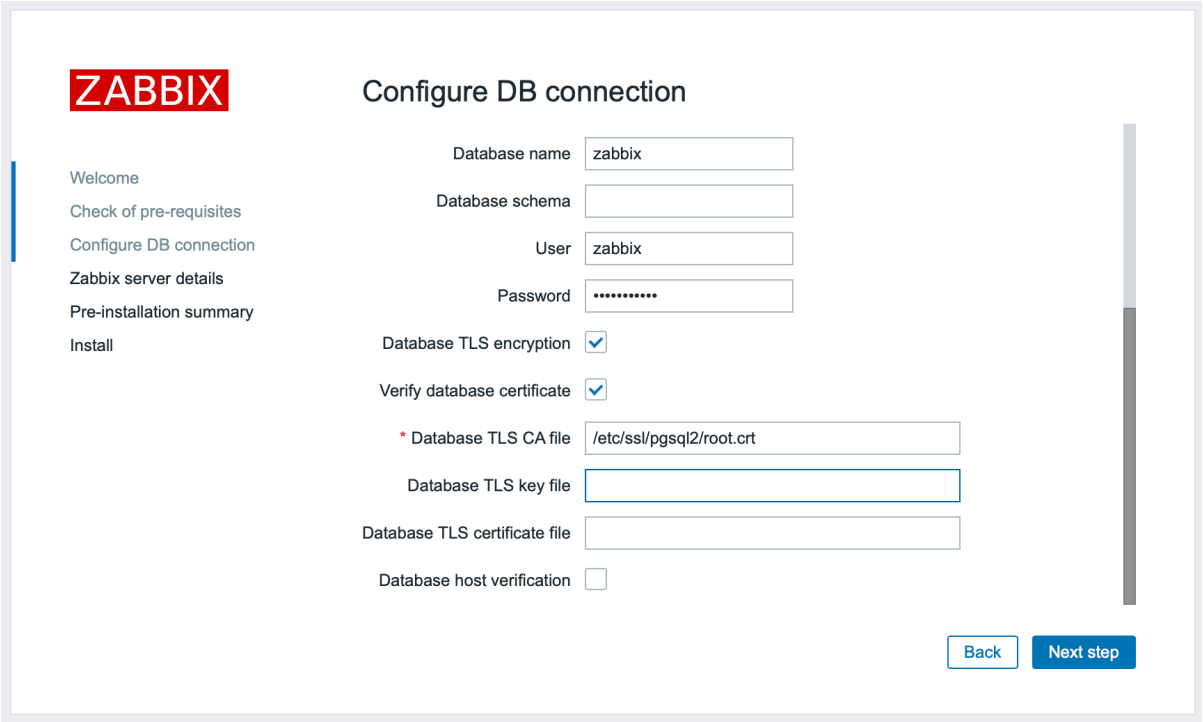
Szyfrowanie z weryfikacją urzędu certyfikacji

Frontend

Aby włączyć szyfrowanie z weryfikacją urzędu certyfikacji dla połączeń między frontend a bazą danych:

- Zaznacz *Szyfrowanie TLS bazy danych* oraz *Weryfikuj certyfikat bazy danych*

- Określ ścieżkę do *pliku CA TLS bazy danych*



Alternatywnie można to ustawić w pliku konfiguracyjnym 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'] = '';
...
```

Serwer

Aby włączyć szyfrowanie z weryfikacją certyfikatu dla połączeń między serwerem Zabbix a bazą danych, edytuj plik konfiguracyjny serwera (zastąp `<strong_password>` silnym, unikalnym hasłem):

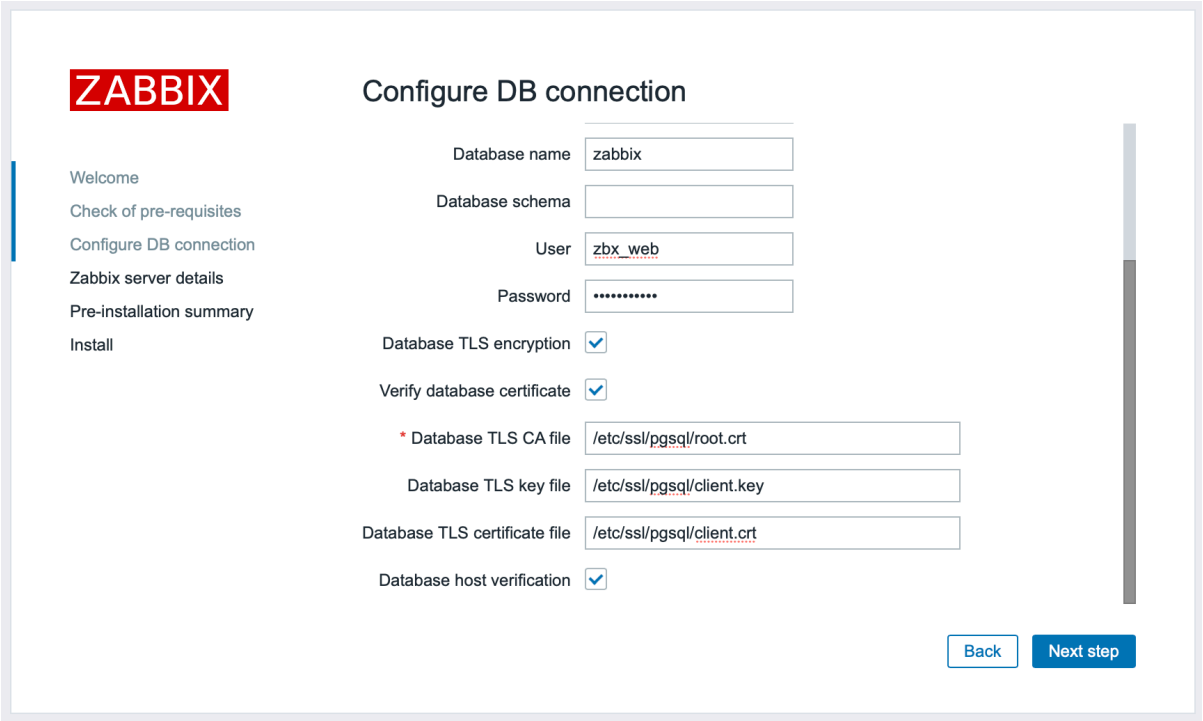
```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/pgsql/root.crt
...
```

Szyfrowanie z pełną weryfikacją

Frontend

Aby włączyć szyfrowanie za pomocą certyfikatu oraz weryfikację tożsamości hosta bazy danych dla połączeń między frontend a bazą danych:

- Zaznacz *Szyfrowanie TLS bazy danych* oraz *Weryfikuj certyfikat bazy danych*
- Określ ścieżkę do *pliku klucza TLS bazy danych*
- Określ ścieżkę do *pliku CA TLS bazy danych*
- Określ ścieżkę do *pliku certyfikatu TLS bazy danych*
- Zaznacz *Weryfikacja hosta bazy danych*



Alternatywnie można to skonfigurować w pliku konfiguracyjnym 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'] = '';
...
```

Serwer

Aby włączyć szyfrowanie z weryfikacją certyfikatu i tożsamości hosta bazy danych dla połączeń między serwerem Zabbix a bazą danych, edytuj plik konfiguracyjny serwera:

```
...
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
...
```

Bezpieczne połączenie między frontendem a serwerem

Przegląd

W tej sekcji przedstawiono kroki konfiguracji Zabbix oraz przykłady konfiguracji bezpiecznych połączeń TLS między frontendem Zabbix a serwerem Zabbix.

Konfiguracja

Domyślnie komunikacja między frontendem Zabbix a serwerem Zabbix jest niezaszyfrowana. Aby zwiększyć bezpieczeństwo, włącz TLS po obu stronach. Poniżej przedstawiono przykład najprostszego sposobu wykonania tej konfiguracji.

1. Wygeneruj certyfikaty i klucze.

Utwórz katalog roboczy:

```
sudo mkdir -p /etc/zabbix/ssl && cd /etc/zabbix/ssl
```

Utwórz certyfikat CA (dostosuj wartość `MyZabbixCA` do rzeczywistej nazwy pospolitej):

```
sudo openssl genrsa -out ca.key 4096
sudo openssl req -new -x509 -days 3650 -key ca.key -out ca.crt -subj "/CN=MyZabbixCA/"
```

Wygeneruj klucz prywatny i certyfikat dla serwera Zabbix (dostosuj wartość `zabbix-server.example.com` do rzeczywistej nazwy pospolitej):

```
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
```

Wygeneruj klucz prywatny i certyfikat dla frontendu Zabbix (dostosuj wartość `zabbix-frontend-node` do rzeczywistej nazwy pospolitej):

```
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. Ustaw odpowiednie uprawnienia.

Dla serwera Zabbix (dostosuj właściciela i grupę zgodnie z użytkownikiem demona serwera Zabbix w swojej dystrybucji):

```
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
```

Dla frontendu (dostosuj właściciela i grupę zgodnie z użytkownikiem serwera WWW w swojej dystrybucji):

```
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. Skonfiguruj serwer Zabbix.

W pliku konfiguracyjnym serwera (`zabbix_server.conf`) dodaj:

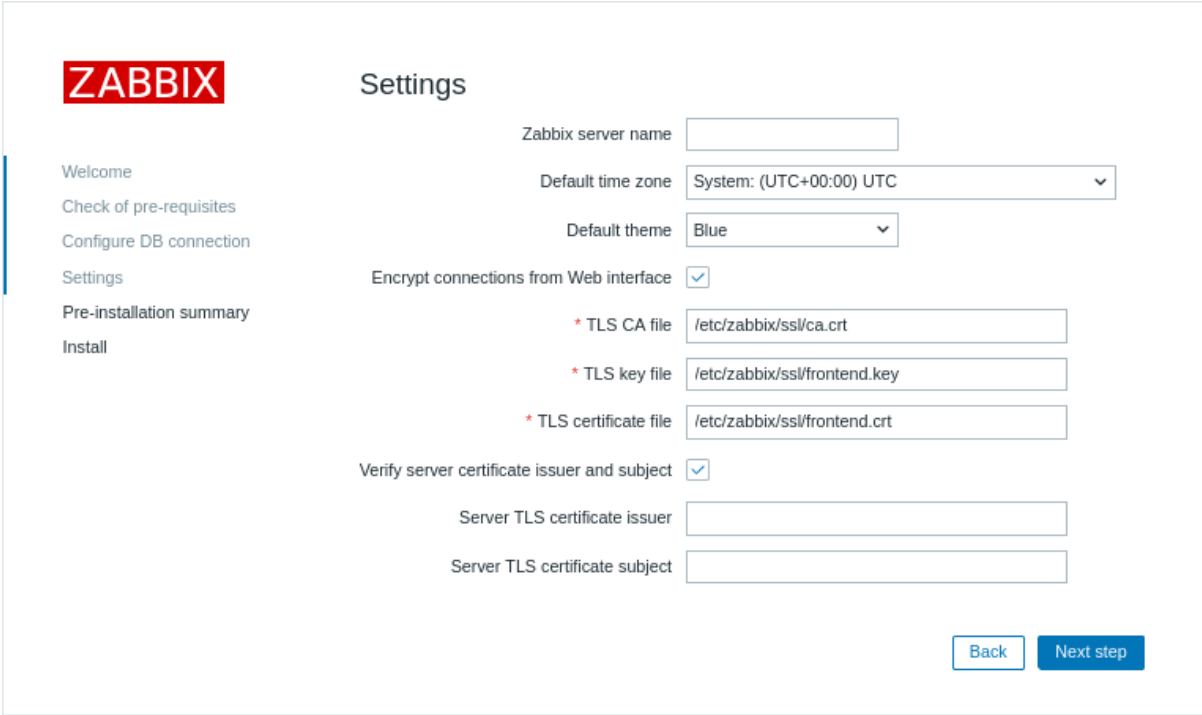
```
TLSFrontendAccept=cert
TLSCertFile=/etc/zabbix/ssl/server.crt
TLSKeyFile=/etc/zabbix/ssl/server.key
TLSCAFile=/etc/zabbix/ssl/ca.crt
### Opcjonalnie:
### TLSFrontendCertIssuer=CN=MyZabbixCA
### TLSFrontendCertSubject=CN=zabbix-frontend-node
```

Następnie uruchom ponownie serwer:

```
sudo systemctl restart zabbix-server
```

4. Skonfiguruj frontend Zabbix.

Podczas instalacji interfejsu internetowego włącz opcję *Szyfruj połączenia z interfejsu internetowego* (oraz, jeśli to konieczne, opcję *Weryfikuj wystawcę i temat certyfikatu serwera*) i wypełnij pola *Plik CA TLS*, *Plik klucza TLS*, *Plik certyfikatu TLS* (oraz, jeśli to konieczne, pola *Wystawca certyfikatu TLS serwera* i *Temat certyfikatu TLS serwera*):



Parametr	Opis
Plik CA TLS	Określ pełną ścieżkę do pliku certyfikatu urzędu certyfikacji (CA) używanego do weryfikacji certyfikatu serwera.
Plik klucza TLS	Określ pełną ścieżkę do pliku prywatnego klucza klienta odpowiadającego certyfikatowi klienta.
Plik certyfikatu TLS	Określ pełną ścieżkę do pliku certyfikatu klienta, jeśli wymagane jest wzajemne uwierzytelnianie TLS.
Wystawca certyfikatu TLS serwera	Określ nazwę wyróżniającą (DN) wystawcy, która ma odpowiadać wystawcy w certyfikacie serwera.
Temat certyfikatu TLS serwera	Określ nazwę wyróżniającą (DN) tematu, która ma odpowiadać tematowi w certyfikacie serwera.

W istniejących instalacjach edytuj następujące parametry w pliku konfiguracyjnym frontendu:

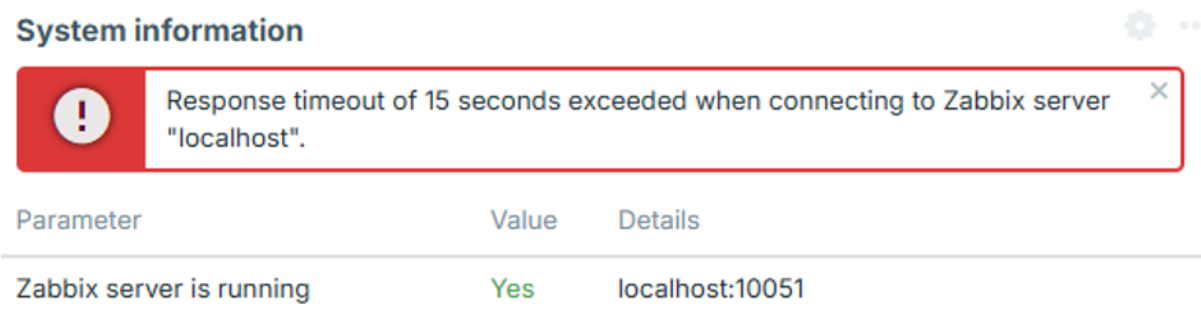
```
$ZBX_SERVER_TLS['ACTIVE'] = 'true';
$ZBX_SERVER_TLS['CA_FILE'] = '/etc/zabbix/ssl/ca.crt';
$ZBX_SERVER_TLS['KEY_FILE'] = '/etc/zabbix/ssl/frontend.key';
$ZBX_SERVER_TLS['CERT_FILE'] = '/etc/zabbix/ssl/frontend.crt';
// Opcjonalnie:
// $ZBX_SERVER_TLS['CERTIFICATE_ISSUER']  = 'CN=MyZabbixCA';
// $ZBX_SERVER_TLS['CERTIFICATE_SUBJECT'] = 'CN=zabbix-server.example.com';
```

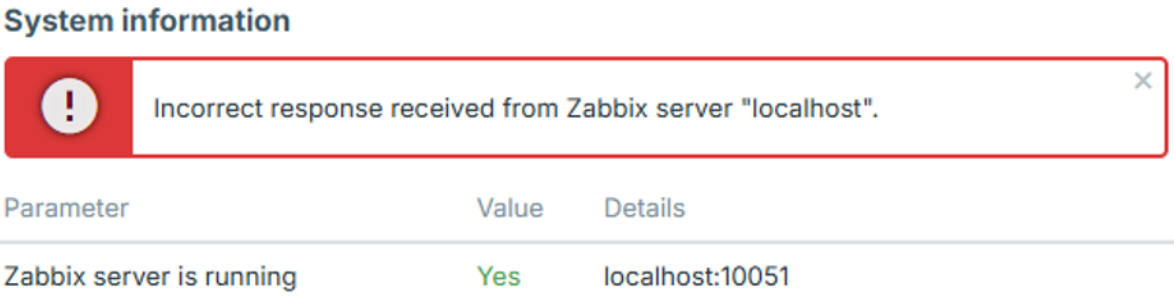
5. Zweryfikuj szyfrowanie, potwierdzając brak komunikatów o błędach w frontendzie Zabbix lub w pliku dziennika serwera Zabbix:

```
tail -f /var/log/zabbix/zabbix_server.log
```

Rozwiązywanie problemów

Nieprawidłowa lub niekompletna konfiguracja TLS może powodować błędy w frontendzie Zabbix lub wskazywać, że serwer Zabbix działa, ale połączenie nie zostało pomyślnie nawiązane:





Aby zidentyfikować dokładny problem z konfiguracją, ustaw poziom dziennika serwera Zabbix na 4, odtwórz błąd we frontendzie i sprawdź dziennik serwera Zabbix pod kątem odpowiedniego komunikatu o błędzie.

Poniższe przypadki opisują typowe błędy i ich możliwe przyczyny.

Adres IP frontend nie jest dozwolony

Dziennik serwera Zabbix zawiera błąd podobny do następującego:

```
54283:20260819:105926.400 frontend connection from "127.0.0.1" is not allowed by FrontendAllowedIP
```

Połączenie jest odrzucane po uzgodnieniu TCP, ponieważ adres IP frontend nie jest autoryzowany.

Dodaj adres IP frontend do parametru `FrontendAllowedIP` w pliku `zabbix_server.conf`, a następnie uruchom ponownie serwer Zabbix.

Połączenie niezaszyfrowane jest niedozwolone

Dziennik serwera Zabbix zawiera błąd podobny do następującego:

```
56884:20260819:111154.285 frontend connection of type "unencrypted" is not allowed by TLSFrontendAccept
```

Serwer Zabbix jest skonfigurowany tak, aby akceptować wyłącznie połączenia TLS, na przykład za pomocą `TLSFrontendAccept=cert`, ale frontend próbuje połączyć się bez szyfrowania.

Włącz szyfrowanie frontendu i skonfiguruj wymagane pliki TLS w `zabbix.conf.php`, zgodnie z opisem w kroku 4 sekcji Konfiguracja.

Frontend próbuje użyć TLS, podczas gdy serwer oczekuje połączenia nieszyfrowanego

W dzienniku serwera Zabbix znajduje się błąd podobny do poniższego:

```
61091:20260819:115916.019 End of zbx_tls_accept():FAIL error:'TLS handshake set result code to 1: file ../
```

Dokładny komunikat o błędzie zależy od wersji OpenSSL.

Serwer Zabbix jest skonfigurowany tak, aby akceptować wyłącznie połączenia nieszyfrowane, na przykład za pomocą ustawienia `TLSFrontendAccept=unencrypted`, podczas gdy frontend próbuje nawiązać połączenie TLS.

Skonfiguruj serwer Zabbix tak, aby akceptował połączenia TLS, ustawiając `TLSFrontendAccept=cert` oraz definiując `TLSCertFile` i `TLSKeyFile` w pliku `zabbix_server.conf`. Alternatywnie wyłącz szyfrowanie frontendu w pliku `zabbix.conf.php`.

Certyfikat frontend nie jest zaufany

W dzienniku serwera Zabbix znajduje się błąd podobny do następującego:

```
62248:20260819:124203.812 End of zbx_tls_accept():FAIL error:'self-signed certificate: TLS handshake set r
```

Frontend przedstawił certyfikat, ale serwer Zabbix nie ufa urzędowi certyfikacji (CA), który go podpisał.

Ten sam problem może również pojawić się w informacjach systemowych frontendu jako:

```
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
```

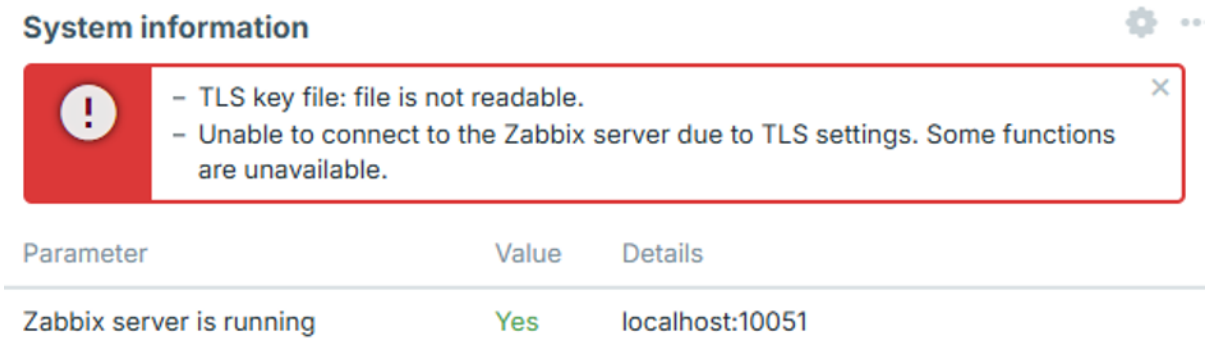
Frontend może również wyświetlać komunikat „Otrzymano nieprawidłową odpowiedź od serwera Zabbix „localhost” ”.

Sprawdź, czy certyfikat frontendu został podpisany przez urząd certyfikacji określony przez `TLSCAFile` w pliku `zabbix_server.conf`.

Plik klucza TLS jest nieczytelny

frontend może wyświetlać komunikat „Plik klucza TLS: plik jest nieczytelny”.

Może również wyświetlać komunikat „Nie można połączyć się z serwerem Zabbix z powodu ustawień TLS. Niektóre funkcje są niedostępne”.

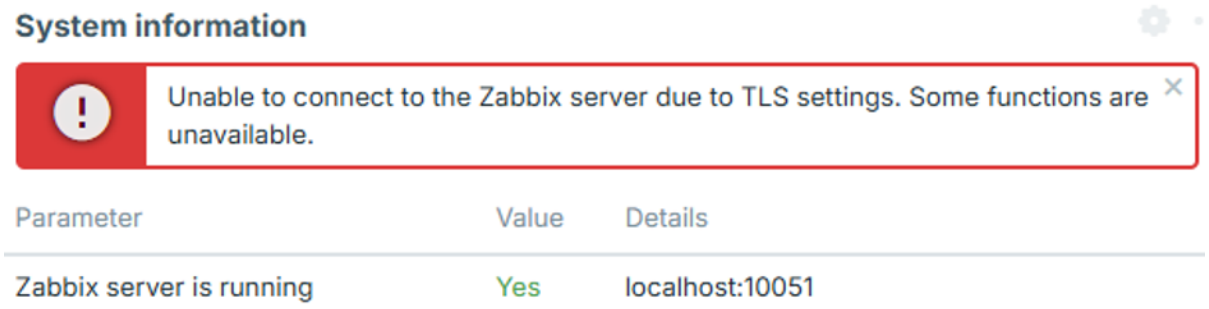


Użytkownik serwera WWW nie ma wystarczających uprawnień do odczytu co najmniej jednego pliku certyfikatu, klucza lub CA skonfigurowanego w `zabbix.conf.php`. Może się tak zdarzyć, gdy pliki certyfikatów są przechowywane w katalogu z ograniczonym dostępem, takim jak /root/ lub katalog domowy użytkownika.

Przenieś pliki certyfikatów do katalogu dostępnego dla serwera WWW i ustaw odpowiednie uprawnienia zgodnie z opisem w kroku 2 sekcji Konfiguracja.

Pliki TLS są nieprawidłowe lub uszkodzone

Frontend może wyświetlać komunikat „Nie można połączyć się z serwerem Zabbix z powodu ustawień TLS. Niektóre funkcje są niedostępne.”.



Może się tak zdarzyć, gdy skonfigurowane pliki certyfikatu lub klucza istnieją i są dostępne do odczytu, ale ich zawartość jest pusta, uszkodzona lub w inny sposób nieprawidłowa. W takim przypadku frontend nie nawiązuje połączenia z serwerem Zabbix, a w dzienniku serwera Zabbix nie pojawia się żadna odpowiadająca mu próba połączenia.

Sprawdź, czy pliki certyfikatu i klucza zawierają prawidłowe dane w odpowiednim formacie oraz czy klucz prywatny odpowiada certyfikatowi.

Kontrola dostępu

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji kontroli dostępu.

Zasada najmniejszych uprawnień

Konta użytkowników powinny przez cały czas działać z możliwie najmniejszymi uprawnieniami. Oznacza to, że konta użytkowników w frontend Zabbix, użytkownicy bazy danych lub użytkownik dla procesów serwer/proxy/agent Zabbix powinni mieć tylko te uprawnienia, które są niezbędne do wykonywania zamierzonych funkcji.

Attention:
Przyznanie dodatkowych uprawnień użytkownikowi ’zabbix’ umożliwi mu dostęp do plików konfiguracyjnych i wykonywanie operacji, które mogą zagrozić bezpieczeństwu infrastruktury.

Podczas konfigurowania uprawnień kont użytkowników należy uwzględnić typy użytkowników frontend Zabbix. Należy pamiętać, że chociaż typ użytkownika *Admin* ma mniej uprawnień niż typ użytkownika *Super Admin*, nadal może zarządzać konfiguracją i wykonywać niestandardowe skrypty.

Note:
Niektóre informacje są dostępne nawet dla użytkowników bez uprawnień. Na przykład, chociaż *Alerts* → *Scripts* jest dostępne tylko dla użytkowników *Super Admin*, skrypty można również pobrać za pośrednictwem Zabbix API. W takim przypadku ograniczenie uprawnień do skryptów i wykluczenie poufnych informacji ze skryptów (na przykład danych dostępowych) może pomóc uniknąć ujawnienia poufnych informacji dostępnych w globalnych skryptach.

Bezpieczny użytkownik dla agent Zabbix

Domyślnie procesy serwer, proxy i agent (lub agent 2) Zabbix współdzielą jednego użytkownika `zabbix`. Aby zapobiec temu, by agent Zabbix/agent 2 (uruchomiony na tym samym komputerze co serwer/proxy) uzyskiwał dostęp do poufnych danych w konfiguracji serwera/proxy (na przykład poświadczeń bazy danych), agent powinien być uruchamiany pod innym użytkownikiem:

Dla agenta Zabbix:

1. Utwórz bezpieczną grupę i użytkownika (np. `zabbix-agent`).
2. Ustaw tego użytkownika w parametrze User w pliku konfiguracyjnym agenta.
3. Uruchom ponownie agent, aby zrzucić uprawnienia do nowego użytkownika.

Dla Zabbix agent 2 konfiguracja musi zostać zastosowana na poziomie usługi, ponieważ plik konfiguracyjny agent 2 nie obsługuje parametru `User`. Przykład można znaleźć w ZBX-26442.

Odebranie dostępu do zapisu do konfiguracji SSL (Windows)

Jeśli skompilowano agent Zabbixa w systemie Windows z OpenSSL znajdującym się w niezabezpieczonym katalogu (np. `C:\zabbix`, `c:\openssl-64bit`, `C:\OpenSSL-Win64-111-static` lub `C:\dev\openssl`), należy upewnić się, że użytkownicy niebędący administratorami nie mają prawa zapisu do tego katalogu. W przeciwnym razie agent wczytuje ustawienia SSL ze ścieżki, którą mogą modyfikować użytkownicy bez uprawnień, co może prowadzić do potencjalnej luki bezpieczeństwa.

Wzmacnianie bezpieczeństwa komponentów Zabbix

Niektóre funkcje można wyłączyć, aby wzmocnić bezpieczeństwo komponentów Zabbix:

- Globalne wykonywanie skryptów na serwerze Zabbix można wyłączyć, ustawiając `EnableGlobalScripts=0` w konfiguracji serwera.
- Globalne wykonywanie skryptów na proxy Zabbix jest domyślnie wyłączone (można je włączyć, ustawiając `EnableRemoteCommands=1` w konfiguracji proxy).
- Globalne wykonywanie skryptów na agentach Zabbix jest domyślnie wyłączone (można je włączyć, dodając parametr `AllowKey=system.run[<command>,*]` lub `AllowKeyRegexp=^system\.run\[.*\]$` dla każdego dozwolonego polecenia w konfiguracji agenta).
- Dostęp do konfiguracji uwierzytelniania HTTP użytkowników jest domyślnie włączony; można go wyłączyć, ustawiając `$ZBX_FEATURE_FLAGS['http_auth_enabled'] = false` w konfiguracji frontend.
- Dostęp do konfiguracji modułów jest domyślnie włączony; można go wyłączyć, ustawiając `$ZBX_FEATURE_FLAGS['modules_config_ = false` w konfiguracji frontend.
- Edytowanie określonych typów mediów można ograniczyć, ustawiając `$ZBX_FEATURE_FLAGS['media_type_denylist'] = [<value1>, <value2>, ...]` w konfiguracji frontend. Możliwe wartości: `'email'`, `'script'`, `'sms'`, `'webhook'` i `'push'`. Można określić jedną lub więcej wartości.

Ponowna instalacja frontend przez uruchomienie `setup.php` jest ograniczona, jeśli istnieje plik konfiguracyjny `zabbix.conf.php`.

Dostęp do ścieżek UNC w systemie Windows przez Zabbix agent

Zabbix agents w systemie Windows obsługują ścieżki UNC (udziały SMB, takie jak `\\server\share\file.txt`) w pozycjach takich jak `vfs.file.*`, `vfs.dir.*`, `modbus.get` i `perf_counter*`. W niektórych kontekstach może to stanowić zagrożenie bezpieczeństwa.

Gdy system Windows otrzyma żądanie dostępu do ścieżki UNC, próbuje uwierzytelnić się na tym serwerze. Oznacza to, że złośliwe żądanie do Zabbix agent może ujawnić skrót NTLM serwerowi żądającego. Użytkownicy mogą ograniczyć to ryzyko, konfigurując w razie potrzeby parametry konfiguracyjne AllowKey, AllowKeyRegexp, DenyKey i DenyKeyRegexp.

Zabezpieczanie MySQL/MariaDB

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące zabezpieczania bazy danych MySQL/MariaDB.

Note:
W przypadku podstawowej konfiguracji zobacz domyślne instrukcje tworzenia bazy danych MySQL/MariaDB, które obejmują utworzenie użytkownika 'zabbix' z pełnymi uprawnieniami do bazy danych Zabbix. Ten użytkownik jest właścicielem bazy danych i ma również niezbędne uprawnienia do modyfikowania struktury bazy danych podczas aktualizacji Zabbix.

Aby poprawić bezpieczeństwo, zaleca się tworzenie dodatkowych ról i użytkowników bazy danych z minimalnymi uprawnieniami. Role i użytkownicy powinni być skonfigurowani zgodnie z zasadą najmniejszych uprawnień, to znaczy powinni mieć wyłącznie uprawnienia niezbędne do wykonywania zamierzonych funkcji.

Tworzenie ról użytkowników

Utwórz następujące role z odpowiednimi uprawnieniami:

- **zbx_srv** - rola do uruchamiania serwera Zabbix i proxy:

```
CREATE ROLE 'zbx_srv';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_srv';
FLUSH PRIVILEGES;
```

- **zbx_web** - rola do uruchamiania frontend Zabbix i API:

```
CREATE ROLE 'zbx_web';
GRANT DELETE, INSERT, SELECT, UPDATE ON zabbix.* TO 'zbx_web';
FLUSH PRIVILEGES;
```

- **zbx_bckp** - rola do tworzenia kopii zapasowej tabel:

```
CREATE ROLE 'zbx_bckp';
GRANT LOCK TABLES, TRIGGER, SELECT ON zabbix.* TO 'zbx_bckp';
GRANT process ON *.* TO 'zbx_bckp';
FLUSH PRIVILEGES;
```

Note:
Przywracanie tabel i aktualizacja powinny być wykonywane przez właściciela bazy danych.

- **zbx_part** - rola z ograniczonym zestawem uprawnień do partycjonowania bazy danych; należy pamiętać, że tę rolę można utworzyć dopiero po utworzeniu bazy danych, ponieważ przyznaje uprawnienia do określonych tabel bazy danych:

```
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;
```

Po utworzeniu ról można je przypisać użytkownikom.

Przypisywanie ról użytkowników

Aby przypisać utworzone role użytkowników, utwórz użytkowników i przypisz im odpowiednie role.
Zastąp <user>, <host>, <role> i <password> odpowiednimi wartościami.
Upewnij się, że <password> jest silne i unikalne.

```
CREATE USER '<user>'@'<host>' IDENTIFIED BY '<password>';
GRANT '<role>' TO '<user>'@'<host>';
SET DEFAULT ROLE '<role>' TO '<user>'@'<host>';
-- For MariaDB: SET DEFAULT ROLE '<role>' FOR '<user>'@'<host>'
FLUSH PRIVILEGES;
```

Na przykład, aby utworzyć i przypisać rolę do uruchamiania serwera Zabbix i 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;
```

Zabezpieczanie PostgreSQL/TimescaleDB

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące zabezpieczania bazy danych PostgreSQL.

> **Note:**
> W przypadku podstawowej konfiguracji zobacz domyślne instrukcje tworzenia bazy danych PostgreSQL, które obejmują utworzenie użytkownika 'zabbix' z pełnymi uprawnieniami do bazy danych Zabbix. Ten użytkownik jest właścicielem bazy danych, który ma również niezbędne uprawnienia do modyfikowania struktury bazy danych podczas aktualizacji Zabbix.

Aby zwiększyć bezpieczeństwo, zaleca się skonfigurowanie bezpiecznego sposobu korzystania ze schematu, a także utworzenie dodatkowych ról i użytkowników bazy danych z minimalnymi uprawnieniami. Role i użytkownicy powinni być skonfigurowani zgodnie z zasadą najmniejszych uprawnień, to znaczy powinni mieć tylko te uprawnienia, które są niezbędne do wykonywania zamierzonych funkcji.

Konfiguracja bazy danych

Utwórz użytkownika, który będzie właścicielem bazy danych, oraz utwórz bazę danych Zabbix; właścicielem bazy danych jest użytkownik określony podczas tworzenia bazy danych:

```
createuser -U postgres -h localhost --pwprompt usr_owner
createdb -U postgres -h localhost -O usr_owner -E Unicode -T template0 zabbix
```

> **Note:**
> Czysta instalacja lub aktualizacja bazy danych musi zostać wykonana przez właściciela bazy danych. Dzieje się tak, ponieważ prawo do usunięcia obiektu bazy danych lub zmiany jego definicji jest uprawnieniem wynikającym z własności bazy danych i nie może zostać przyznane ani odebrane.

> **Attention:**
> Poniższe polecenia na tej stronie muszą zostać wykonane podczas połączenia z PostgreSQL ustanowionego konkretnie z bazą danych `zabbix`.

Utwórz schemat `zabbix` i ustaw właściciela bazy danych (`usr_owner`) jako właściciela tego schematu:

```
CREATE SCHEMA zabbix AUTHORIZATION usr_owner;
```

Skonfiguruj bezpieczny wzorzec użycia schematu:

```
REVOKE CREATE ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON DATABASE zabbix FROM PUBLIC;
-- Uwaga: search_path powinien wskazywać na schemat "zabbix":
ALTER ROLE ALL IN DATABASE zabbix SET search_path = "zabbix";
```

Po skonfigurowaniu bazy danych przejdź do tworzenia ról użytkowników.

Tworzenie ról użytkowników

Utwórz następujące role z odpowiednimi uprawnieniami:

- **zbx_srv** - rola do uruchamiania serwera Zabbix i 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** - rola do uruchamiania frontend Zabbix i 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** - rola do tworzenia kopii zapasowej tabel:

```
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:
Przywracanie tabel jest możliwe tylko przez właściciela bazy danych.

Po utworzeniu ról można je przypisać użytkownikom.

Przypisywanie ról użytkowników

Aby przypisać utworzone role użytkowników, utwórz użytkowników i przypisz im odpowiednie role.
Zastąp <user>, <role> i <password> odpowiednimi wartościami.
Upewnij się, że <password> jest silne i unikalne.

```
CREATE USER <user> WITH ENCRYPTED password '<password>';
GRANT <role> TO <user>;
```

Na przykład, aby utworzyć i przypisać rolę do uruchamiania serwera Zabbix i proxy:

```
CREATE USER usr_srv WITH ENCRYPTED password 'strong-password';
GRANT zbx_srv TO usr_srv;
```

Partycjonowanie bazy danych z TimescaleDB

Partycjonowanie bazy danych jest obsługiwane przez TimescaleDB. Aby korzystać z TimescaleDB, serwer Zabbix wymaga uprawnień właściciela bazy danych.

Jeśli schemat PostgreSQL zabbix został już utworzony w bazie danych zabbix, można włączyć TimescaleDB za pomocą następującego polecenia:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb WITH SCHEMA zabbix CASCADE;" | sudo -u postgres psql zabb
```

Kryptografia

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji kryptografii.

Konfigurowanie SSL dla frontend Zabbix

W systemach opartych na RHEL zainstaluj pakiet mod_ssl:

```
dnf install mod_ssl
```

Utwórz katalog dla kluczy SSL:

```
mkdir -p /etc/httpd/ssl/private
chmod 700 /etc/httpd/ssl/private
```

Utwórz certyfikat SSL:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/httpd/ssl/private/apache-selfsigned.key -
```

Wypełnij odpowiednio wyświetlane monity. Najważniejsza jest linia, która prosi o Common Name. Musisz podać nazwę domeny, która ma być powiązana z twoim serwerem. Jeśli nie masz nazwy domeny, możesz zamiast niej podać publiczny adres IP.

```
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 []:
```

Edytuj plik konfiguracji SSL 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
```

Uruchom ponownie usługę Apache, aby zastosować zmiany:

```
systemctl restart httpd.service
```

Zabezpieczanie serwera WWW

Przegląd

Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji serwera WWW.

Wymuszanie przekierowania adresu URL root do Zabbix SSL

W systemach opartych na RHEL dodaj wirtualny host do konfiguracji Apache (`/etc/httpd/conf/httpd.conf`) i ustaw trwałe przekierowanie katalogu głównego dokumentów na adres URL Zabbix SSL. Zwróć uwagę, że `example.com` należy zastąpić rzeczywistą nazwą serwera.

```
### Dodaj wiersze:

<VirtualHost *:*>
    ServerName example.com
    Redirect permanent / https://example.com
</VirtualHost>
```

Uruchom ponownie usługę Apache, aby zastosować zmiany:

```
systemctl restart httpd.service
```

Włączanie HTTP Strict Transport Security (HSTS) na serwerze WWW

Aby chronić frontend Zabbixa przed atakami polegającymi na obniżeniu wersji protokołu, zalecamy włączenie polityki HSTS na serwerze WWW.

Aby włączyć politykę HSTS dla swojego frontendu Zabbixa w konfiguracji Apache, wykonaj następujące kroki:

1. Znajdź plik konfiguracyjny swojego wirtualnego hosta:

- `/etc/httpd/conf/httpd.conf` w systemach opartych na RHEL
- `/etc/apache2/sites-available/000-default.conf` w Debianie/Ubuntu

2. Dodaj następującą dyrektywę do pliku konfiguracyjnego swojego wirtualnego hosta:

```
<VirtualHost *:*>
    Header set Strict-Transport-Security "max-age=31536000"
</VirtualHost>
```

3. Uruchom ponownie usługę Apache, aby zastosować zmiany:

```
### W systemach opartych na RHEL:
systemctl restart httpd.service

### W Debianie/Ubuntu
systemctl restart apache2.service
```

Wymuszanie bezpiecznych i SameSite ciasteczek sesji w Zabbix

Podczas konfiguracji Zabbix istotne jest wymuszenie atrybutów secure i SameSite dla ciasteczek sesji, aby zwiększyć bezpieczeństwo i zapobiec atakom typu cross-site request forgery (CSRF). Jednak wymuszenie `SameSite=Strict` może powodować problemy w niektórych scenariuszach, takich jak:

- Widżety URL na pulpicie pokazujące komunikat „user not logged in" podczas osadzania ramek iframe z tej samej domeny.
- Użytkownicy uzyskujący dostęp do pulpitu przez HTTP zamiast HTTPS mogą napotkać problemy z logowaniem.
- Brak możliwości udostępniania adresów URL do określonych sekcji menu Zabbix lub hostów.

Aby ograniczyć te problemy, użytkownicy powinni mieć możliwość dostosowania polityki SameSite.

1. Bezpieczne ciasteczka

Ustawienie flagi `secure` zapewnia, że ciasteczka są przesyłane wyłącznie przez HTTPS, co zapobiega ich ujawnieniu przez nieszyfrowane połączenia.

Aby włączyć bezpieczne ciasteczka w Zabbix, dodaj lub zmodyfikuj następujące ustawienie w konfiguracji serwera WWW:

Dla Apache:

```
Header always edit Set-Cookie ^(.*)$ $1;Secure
```

Dla Nginx:

```
proxy_cookie_path / "/; Secure";
```

Upewnij się, że frontend Zabbix jest dostępny przez HTTPS; w przeciwnym razie ciasteczka z flagą `Secure` nie będą wysyłane.

2. Konfigurowanie atrybutu SameSite

Ustawienia serwera WWW mogą również wymuszać atrybut SameSite:

Dla Apache:

```
<IfModule mod_headers.c>
    Header onsuccess edit Set-Cookie (.*) "$1; SameSite=Strict"
</IfModule>
```

Dla Nginx (wersja 1.19.3+):

```
proxy_cookie_flags ~ samesite=Strict; # Replace ~ with 'zbx_session' for specificity
```

Włączanie Content Security Policy (CSP) na serwerze WWW

Aby chronić frontend Zabbix przed Cross Site Scripting (XSS), wstrzykiwaniem danych i podobnymi atakami, zalecamy włączenie Content Security Policy na serwerze WWW. W tym celu skonfiguruj serwer WWW tak, aby zwracał nagłówek HTTP.

Attention:
Poniższa konfiguracja nagłówka CSP dotyczy wyłącznie domyślnej instalacji frontend Zabbix oraz przypadków, gdy cała zawartość pochodzi z domeny witryny (z wyłączeniem subdomen). Inna konfiguracja nagłówka CSP może być wymagana, jeśli na przykład konfigurujesz widżet *URL* do wyświetlania zawartości z subdomen witryny lub domen zewnętrznych, przełączasz się z *OpenStreetMap* na inny silnik map, albo dodajesz zewnętrzny CSS lub widżety. Jeśli używasz metody uwierzytelniania Duo Universal Prompt wieloskładnikowego, upewnij się, że dodasz "duo.com" do dyrektywy CSP w pliku konfiguracyjnym wirtualnego hosta.

Aby włączyć CSP dla frontend Zabbix w konfiguracji Apache, wykonaj następujące kroki:

1. Znajdź plik konfiguracyjny swojego wirtualnego hosta:

- `/etc/httpd/conf/httpd.conf` w systemach opartych na RHEL
- `/etc/apache2/sites-available/000-default.conf` w Debian/Ubuntu

2. Dodaj następującą dyrektywę do pliku konfiguracyjnego swojego wirtualnego hosta:

```
<VirtualHost *:*>
    Header set Content-Security-Policy: "default-src 'self' *.openstreetmap.org; script-src 'self' 'unsafe
</VirtualHost>
```

3. Uruchom ponownie usługę Apache, aby zastosować zmiany:

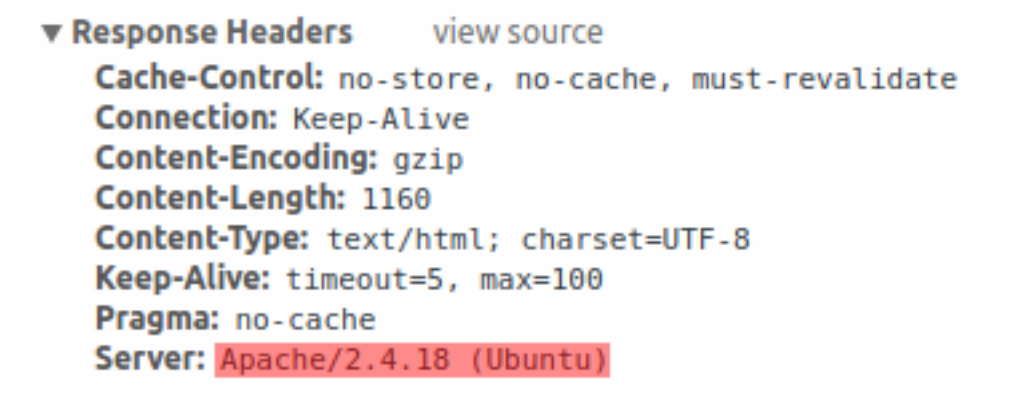
```
### On RHEL-based systems:
systemctl restart httpd.service

### On Debian/Ubuntu
systemctl restart apache2.service
```

Wyłączanie ujawniania informacji o serwerze WWW

W celu zwiększenia bezpieczeństwa zaleca się wyłączenie wszystkich sygnatur serwera WWW.

Domyślnie serwer WWW ujawnia sygnaturę oprogramowania:



Sygnaturę można wyłączyć, dodając następujące parametry do pliku konfiguracyjnego Apache:

```
ServerSignature Off
ServerTokens Prod
```

Sygnaturę PHP (nagłówek HTTP X-Powered-By) można wyłączyć, zmieniając plik konfiguracyjny `php.ini` (domyślnie sygnatura jest wyłączona):

```
expose_php = Off
```

Aby zmiany w pliku konfiguracyjnym zostały zastosowane, wymagane jest ponowne uruchomienie serwera WWW.

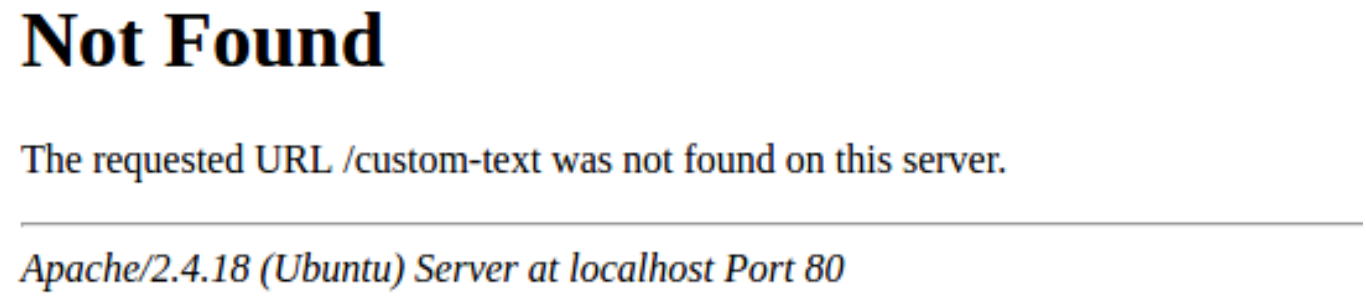
W celu dodatkowego zwiększenia bezpieczeństwa można użyć narzędzia *mod_security* z Apache (pakiet *libapache2-mod-security2*). Narzędzie to umożliwia usunięcie sygnatury serwera, a nie tylko usunięcie z niej numeru wersji. Sygnaturę serwera można zmienić na dowolną wartość, ustawiając parametr „SecServerSignature” na wybraną wartość po zainstalowaniu *mod_security*.

Informacje dotyczące usuwania lub zmiany sygnatur oprogramowania można znaleźć w dokumentacji używanego serwera WWW.

Wyłączanie domyślnych stron błędów serwera WWW

Aby uniknąć ujawnienia informacji, zaleca się wyłączenie domyślnych stron błędów.

Domyślnie serwer WWW używa wbudowanych stron błędów:



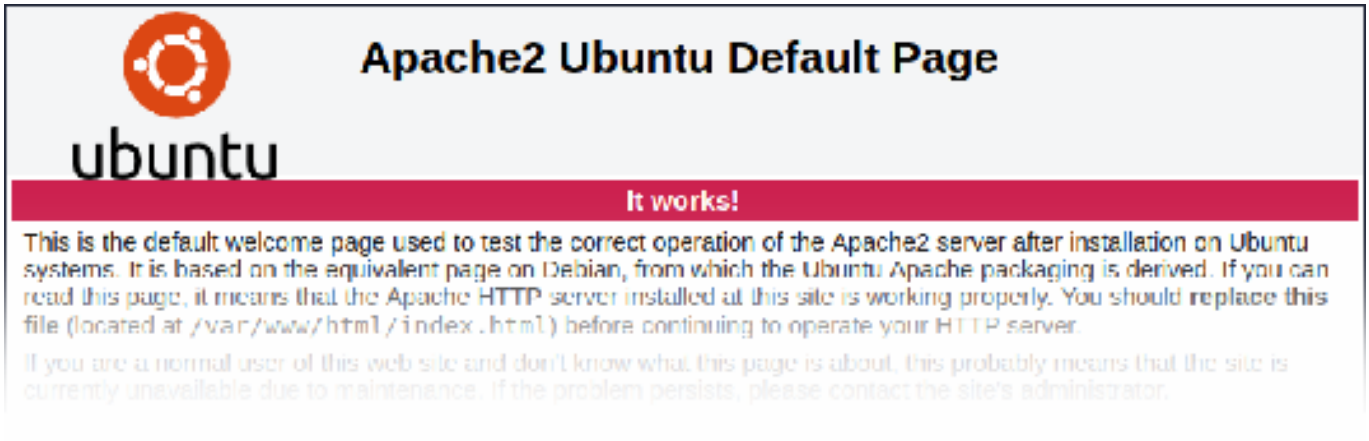
Te domyślne strony błędów należy zastąpić lub usunąć. Na przykład dyrektywa „ErrorDocument” może służyć do zdefiniowania niestandardowej strony lub tekstu błędu dla serwera WWW Apache.

Informacje na temat zastępowania lub usuwania domyślnych stron błędów można znaleźć w dokumentacji używanego serwera WWW.

Usuwanie testowej strony serwera WWW

Aby uniknąć ujawnienia informacji, zaleca się usunięcie testowej strony serwera WWW.

Domyślnie katalog główny serwera WWW Apache zawiera testową stronę `index.html`:



Informacje dotyczące usuwania domyślnych stron testowych można znaleźć w dokumentacji używanego serwera WWW.

Ustaw nagłówek odpowiedzi HTTP X-Frame-Options

Domyślnie Zabbix jest skonfigurowany z parametrem *Use X-Frame-Options HTTP header* ustawionym na `SAMEORIGIN`. Oznacza to, że zawartość może być ładowana tylko w ramce, która ma to samo pochodzenie co sama strona.

Elementy frontend Zabbix, które pobierają zawartość z zewnętrznych adresów URL (a konkretnie widżet dashboard widget URL), wyświetlają pobraną zawartość w piaskownicy z włączonymi wszystkimi ograniczeniami piaskownicy.

Te ustawienia zwiększają bezpieczeństwo frontend Zabbix i zapewniają ochronę przed atakami XSS oraz clickjacking. Użytkownicy *Super admin* mogą w razie potrzeby modyfikować parametry *Use iframe sandboxing* oraz *Use X-Frame-Options HTTP header*. Przed zmianą ustawień domyślnych należy dokładnie rozważyć ryzyko i korzyści. Całkowite wyłączenie piaskownicy iframe lub nagłówka HTTP X-Frame-Options nie jest zalecane.

Ukrywanie pliku z listą popularnych haseł

Aby zwiększyć złożoność ataków brute force na hasła, zaleca się ograniczenie dostępu do pliku `ui/data/top_passwords.txt`. Plik ten zawiera listę najczęściej używanych oraz zależnych od kontekstu haseł i uniemożliwia użytkownikom ustawianie takich haseł (jeśli parametr *Avoid easy-to-guess passwords* jest włączony w polityce haseł).

Aby ograniczyć dostęp do pliku `top_passwords.txt`, zmodyfikuj konfigurację swojego serwera WWW.

W Apache dostęp do pliku można ograniczyć za pomocą pliku `.htaccess`:

```
<Files "top_passwords.txt">
    Order Allow,Deny
    Deny from all
</Files>
```

W NGINX dostęp do pliku można ograniczyć za pomocą dyrektywy `location`:

```
location = /data/top_passwords.txt {
    deny all;
    return 404;
}
```

Przechowywanie sekretów

Przegląd Zabbix można skonfigurować tak, aby pobierał poufne informacje z bezpiecznego sejfu. Obsługiwane są następujące usługi zarządzania sekretami: HashiCorp Vault KV Secrets Engine - Version 2, CyberArk Vault CV12.

Sekrety mogą być używane do pobierania:

- wartości makr użytkownika
- poświadczeń dostępu do bazy danych

Zabbix zapewnia dostęp tylko do odczytu do sekretów w sejfie, zakładając, że sekrety są zarządzane przez inną osobę.

Aby uzyskać informacje o konfiguracji konkretnego dostawcy sejfu, zobacz:

- Konfiguracja HashiCorp
- Konfiguracja CyberArk

Buforowanie wartości sekretów Domyślnie wartości makr sekretów z vault są pobierane przez serwer Zabbix przy każdym odświeżeniu danych konfiguracyjnych, a następnie przechowywane w pamięci podręcznej konfiguracji. proxy Zabbix otrzymuje wartości makr sekretów z vault od serwera Zabbix przy każdej synchronizacji konfiguracji i przechowuje je we własnej pamięci podręcznej konfiguracji.

> **Attention:**
> Szyfrowanie między serwerem Zabbix a proxy musi być włączone; w przeciwnym razie w dzienniku zostanie zapisane ostrzeżenie serwera.

Można również skonfigurować, aby wartości makr były pobierane niezależnie przez serwer Zabbix i proxy Zabbix.

Aby ręcznie wyzwolić odświeżenie wartości sekretów z vault przechowywanych w pamięci podręcznej, użyj opcji wiersza poleceń `secrets_reload` option.

Buforowanie danych uwierzytelniających bazy danych frontend Zabbix jest domyślnie wyłączone, ale można je włączyć, ustawiając opcję `$DB['VAULT_CACHE'] = true` w pliku konfiguracyjnym frontend. Dane uwierzytelniające będą przechowywane w lokalnej pamięci podręcznej z użyciem katalogu plików tymczasowych systemu plików. Serwer WWW musi zezwalać na zapis w prywatnym katalogu tymczasowym (na przykład w przypadku Apache należy ustawić opcję konfiguracyjną `PrivateTmp=True`). Aby określić, jak często pamięć podręczna danych ma być odświeżana/unieważniana, użyj stałej `ZBX_DATA_CACHE_TTL`.

Konfiguracja TLS Aby skonfigurować TLS dla komunikacji między komponentami Zabbix a sejfem, dodaj certyfikat podpisany przez urząd certyfikacji (CA) do systemowego domyślnego magazynu CA. Aby użyć innej lokalizacji, określ katalog w parametrze konfiguracyjnym SSLCALocation Zabbix serwer/proxy, umieść plik certyfikatu w tym katalogu, a następnie uruchom polecenie CLI:

```
c_rehash .
```

Konfiguracja CyberArk

Ta sekcja wyjaśnia, jak skonfigurować Zabbix do pobierania sekretów z CyberArk Vault CV12.

Vault powinien zostać zainstalowany i skonfigurowany zgodnie z opisem w oficjalnej dokumentacji CyberArk.

Aby dowiedzieć się, jak skonfigurować TLS w Zabbix, zobacz *Przechowywanie sekretów*.

Dane uwierzytelniające bazy danych

Dostęp do sekretu zawierającego dane uwierzytelniające bazy danych konfiguruje się osobno dla każdego komponentu Zabbix.

Serwer i proxy

Aby pobierać dane uwierzytelniające bazy danych z magazynu sekretów dla serwera Zabbix lub proxy, określ w pliku konfiguracyjnym następujące parametry:

- `Vault` - dostawca magazynu sekretów, który ma być używany;
- `VaultURL` - adres URL serwera magazynu sekretów w protokole HTTP(S);
- `VaultDBPath` - zapytanie do sekretu w magazynie sekretów zawierającego dane uwierzytelniające bazy danych, które zostaną pobrane za pomocą kluczy „Content” i „UserName” (tej opcji można używać tylko wtedy, gdy nie określono DBUser i DBPassword);
- `VaultTLSCertFile`, `VaultTLSKeyFile` - nazwy plików certyfikatu SSL i klucza; skonfigurowanie tych opcji nie jest wymagane, ale jest zdecydowanie zalecane;
- `VaultPrefix` - niestandardowy prefiks ścieżki lub zapytania magazynu sekretów, zależnie od magazynu; jeśli nie zostanie określony, zostanie użyta najbardziej odpowiednia wartość domyślna.

> **Attention:**
> Parametry konfiguracyjne `Vault`, `VaultURL`, `VaultTLSCertFile`, `VaultTLSKeyFile` i `VaultPrefix` są również używane do uwierzytelniania w magazynie sekretów podczas przetwarzania makr sekretów magazynu przez serwer Zabbix (oraz proxy Zabbix, jeśli zostało skonfigurowane). Serwer Zabbix i proxy nie otworzą makr sekretów magazynu zawierających dane uwierzytelniające bazy danych z VaultDBPath.

Serwer Zabbix i proxy Zabbix odczytują parametry konfiguracyjne związane z magazynem sekretów z plików `zabbix_server.conf` i `zabbix_proxy.conf` podczas uruchamiania.

Przykład

1. W pliku `zabbix_server.conf` określ następujące parametry:

```
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 wyśle do vault następujące żądanie API:

```
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. Odpowiedź vault będzie zawierać klucze ”Content” i ”UserName”:

```
{
    "Content": <password>,
    "UserName": <username>,
    "Address": <address>,
    "Database": <Database>,
    "PasswordChangeInProcess":<PasswordChangeInProcess>
}
```

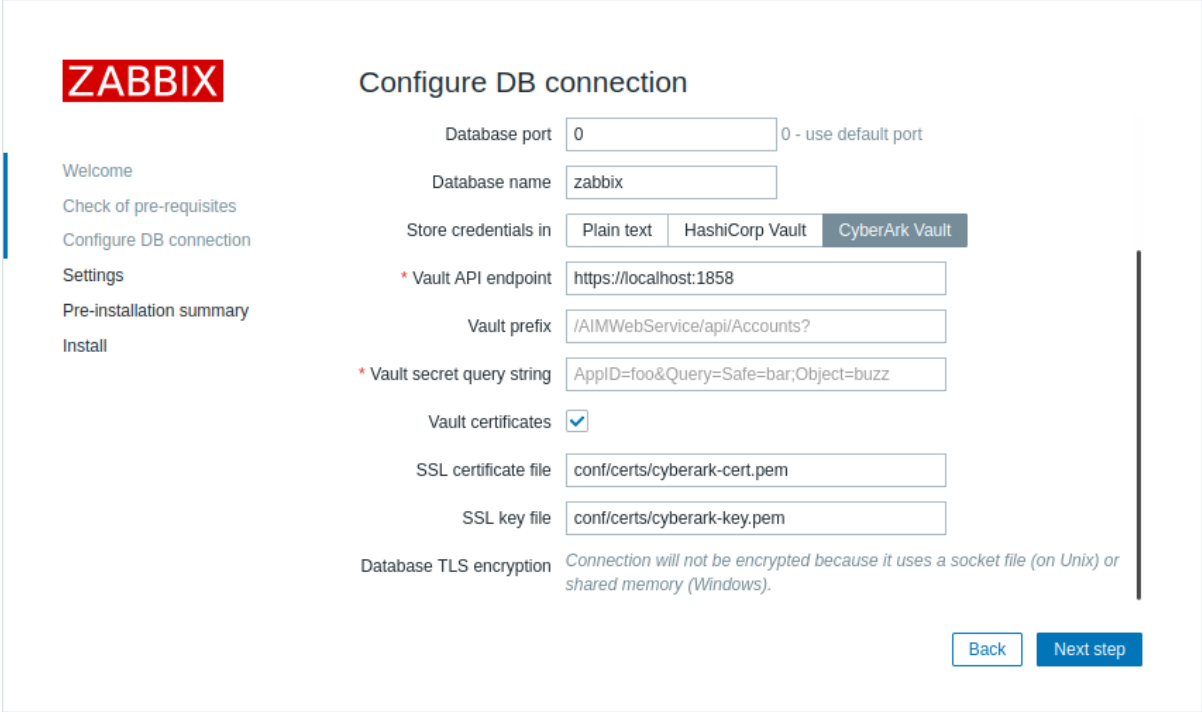
4. W rezultacie Zabbix użyje następujących poświadczeń do uwierzytelniania w bazie danych:

- Nazwa użytkownika: <username>
- Hasło: <password>

frontend

Aby uzyskać dane uwierzytelniające do bazy danych z sejfu dla frontend Zabbix, określ następujące parametry podczas instalacji frontend.

1. W kroku *Configure DB Connection* ustaw parametr *Store credentials in* na *CyberArk Vault*.



2. Następnie uzupełnij dodatkowe parametry:

Parameter	Mandatory	Default value	Description
Vault API endpoint	yes	https://localhost:1858	Określ adres URL połączenia z sejfem w formacie `scheme://host:port`

Parameter	Mandatory	Default value	Description
Vault prefix	no	/AIMWebService/api/Accounts?	Podaj niestandardowy prefiks ścieżki lub zapytania sejfu. Jeśli nie zostanie określony, zostanie użyta wartość domyślna.
Vault secret query string	yes		Zapytanie określające, skąd powinny zostać pobrane dane uwierzytelniające do bazy danych. Przykład: `AppID=foo&Query=Safe=bar;Object=buzz`
Vault certificates	no		Po zaznaczeniu pola wyboru pojawią się dodatkowe parametry umożliwiające skonfigurowanie uwierzytelniania klienta. Chociaż ten parametr jest opcjonalny, zdecydowanie zaleca się jego włączenie na potrzeby komunikacji z CyberArk Vault.
SSL certificate file	no	conf/certs/cyberark-cert.pem	Ścieżka do pliku certyfikatu SSL. Plik musi być w formacie PEM. Jeśli plik certyfikatu zawiera również klucz prywatny, pozostaw parametr SSL key file pusty.
SSL key file	no	conf/certs/cyberark-key.pem	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta. Plik musi być w formacie PEM.

Wartości makr użytkownika

Aby używać CyberArk Vault do przechowywania wartości makr użytkownika *Vault secret*, upewnij się, że:

- serwer Zabbix jest skonfigurowany do pracy z CyberArk Vault;
- parametr *Vault provider* w *Administration > General > Other* jest ustawiony na "CyberArk Vault".



Note:
Serwer Zabbix (oraz proxy Zabbix, jeśli jest skonfigurowany) wymagają dostępu do wartości makr *Vault secret* z vault. Frontend Zabbix nie potrzebuje takiego dostępu.

Wartość makra powinna zawierać zapytanie (w formacie `query:key`).

Szczegółowe informacje o przetwarzaniu wartości makr przez Zabbix znajdziesz w sekcji *Vault secret macros*.

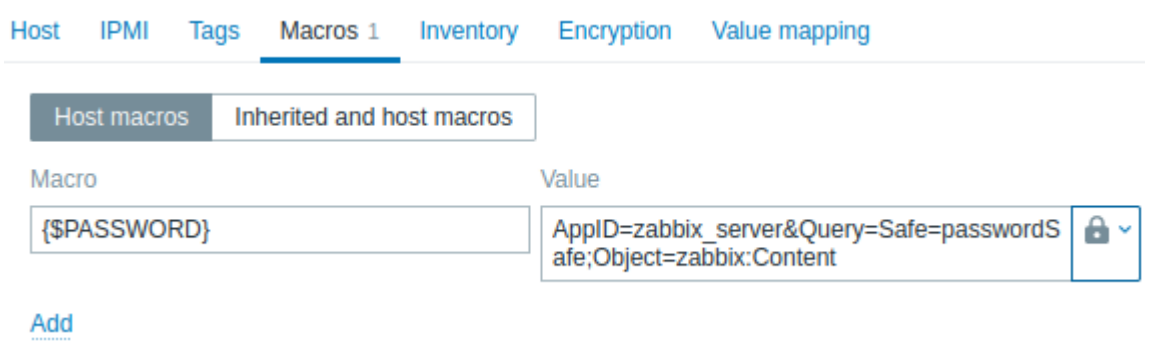
Składnia zapytania

Symbol dwukropka (`:`) jest zarezerwowany do oddzielania zapytania od klucza.

Jeśli zapytanie zawiera ukośnik lub dwukropek, symbole te należy zakodować w adresie URL (`/` jest kodowany jako `%2F`, a `:` jako `%3A`).

Przykład

1. W Zabbix dodaj makro użytkownika {$PASSWORD} typu *Vault secret* i ustaw jego wartość na `AppID=zabbix_server&Query=Safe=pa`



2. Zabbix wyśle do magazynu następujące żądanie API:

```
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. Odpowiedź magazynu będzie zawierać klucz `Content`:

```
{
    "Content": <password>,
    "UserName": <username>,
    "Address": <address>,
    "Database" :<Database>,
    "PasswordChangeInProcess":<PasswordChangeInProcess>
}
```

4. W rezultacie Zabbix rozwiąże makro `{$PASSWORD}` do wartości - `<password>`

Aktualizowanie istniejącej konfiguracji

Aby zaktualizować istniejącą konfigurację pobierania sekretów z CyberArk Vault:

1. Zaktualizuj parametry pliku konfiguracyjnego serwera Zabbix lub proxy zgodnie z opisem w sekcji *Dane uwierzytelniające bazy danych*.
2. Zaktualizuj parametry pliku konfiguracyjnego frontendu 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. W razie potrzeby skonfiguruj makra użytkownika zgodnie z opisem w sekcji *Wartości makr użytkownika*.

Aby zaktualizować istniejącą konfigurację pobierania sekretów z HashiCorp Vault, zobacz sekcję *Konfiguracja HashiCorp*.

Konfiguracja HashiCorp

Przegląd

Ta sekcja wyjaśnia, jak skonfigurować Zabbix do pobierania sekretów z HashiCorp Vault KV Secrets Engine - Version 2.

Vault powinien być wdrożony i skonfigurowany zgodnie z opisem w oficjalnej dokumentacji HashiCorp.

Aby dowiedzieć się, jak skonfigurować TLS w Zabbix, zobacz *Przechowywanie sekretów*.

Pobieranie danych uwierzytelniających do bazy danych

Aby pomyślnie pobrać sekret zawierający dane uwierzytelniające do bazy danych, należy skonfigurować oba elementy:

- serwer/proxy Zabbix
- frontend Zabbix

Serwer/proxy

Aby skonfigurować serwer lub proxy Zabbix, określ następujące parametry konfiguracji w pliku konfiguracyjnym:

- `Vault` - dostawca magazynu sekretów, który ma być używany.
- `VaultToken` - token uwierzytelniania magazynu sekretów (szczegółowe informacje znajdziesz w pliku konfiguracyjnym serwera/proxy Zabbix); nie można go używać razem z `VaultAppRoleID`.
- `VaultAppRoleID` i `VaultAppSecretID` - identyfikator roli i identyfikator sekretu używane do uwierzytelniania w magazynie sekretów (szczegółowe informacje znajdziesz w pliku konfiguracyjnym serwera/proxy Zabbix); nie można ich używać razem z `VaultToken`.
- `VaultURL` - adres URL HTTP(S) serwera magazynu sekretów.
- `VaultDBPath` - ścieżka do sekretu w magazynie sekretów zawierającego dane uwierzytelniające do bazy danych (tej opcji można używać tylko wtedy, gdy nie określono DBUser i DBPassword). Serwer lub proxy Zabbix pobierze dane uwierzytelniające za pomocą kluczy „password” i „username”.

- `VaultPrefix` - niestandardowy prefiks ścieżki lub zapytania magazynu sekretów, zależnie od magazynu; jeśli nie zostanie określony, zostanie użyta najbardziej odpowiednia wartość domyślna. Pamiętaj, że prefiks musi zaczynać się i kończyć znakiem /.

Attention:
Parametry konfiguracji `Vault`, `VaultToken` (lub `VaultAppRoleID`/`VaultAppSecretID`), `VaultURL` i `VaultPrefix` są również używane do uwierzytelniania w magazynie sekretów podczas przetwarzania makr magazynu sekretów przez serwer Zabbix (oraz proxy Zabbix, jeśli zostało skonfigurowane). Serwer Zabbix i proxy nie otworzą makr magazynu sekretów zawierających dane uwierzytelniające do bazy danych z VaultDBPath.

Zdecydowanie zaleca się używanie różnych tokenów dla różnych proxy.

Serwer Zabbix i proxy Zabbix odczytują parametry konfiguracji związane z magazynem sekretów z plików `zabbix_server.conf` i `zabbix_proxy.conf` podczas uruchamiania. Dodatkowo serwer Zabbix i proxy Zabbix odczytają zmienną środowiskową `VAULT_TOKEN` jednokrotnie podczas uruchamiania i usuną ją, aby nie była dostępna za pośrednictwem skryptów uruchamianych przez forkowanie; wystąpienie wartości zarówno w parametrach `VaultToken`, jak i `VAULT_TOKEN` jest błędem.

Przykład

1. W pliku `zabbix_server.conf` określ następujące parametry.

Korzystając z VaultToken:

```
Vault=HashiCorp
VaultToken=hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTYONFNSaUp3ZnpWbDF1RUNjUkNTZEg
VaultURL=https://127.0.0.1:8200
VaultDBPath=database
VaultPrefix=/v1/secret/data/zabbix/
```

Lub korzystając z 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. Uruchom następujące polecenia CLI, aby utworzyć wymagany sekret w vault:

```
#### Włącz punkt montowania "secret/", jeśli nie jest jeszcze włączony; pamiętaj, że należy użyć "kv-v2".
vault secrets enable -path=secret/ kv-v2

#### Dodaj nowe sekrety z kluczami username i password w punkcie montowania "secret/" oraz ścieżce "zabbix
vault kv put -mount=secret zabbix/database username=zabbix password=<password>

#### Sprawdź, czy sekret został pomyślnie dodany.
vault kv get secret/zabbix/database

#### Na koniec przetestuj za pomocą Curl; pamiętaj, że "data" należy ręcznie dodać po punkcie montowania,
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/database
```

3. W rezultacie serwer Zabbix pobierze następujące poświadczenia do uwierzytelniania w bazie danych:

- Nazwa użytkownika: zabbix
- Hasło: <password>

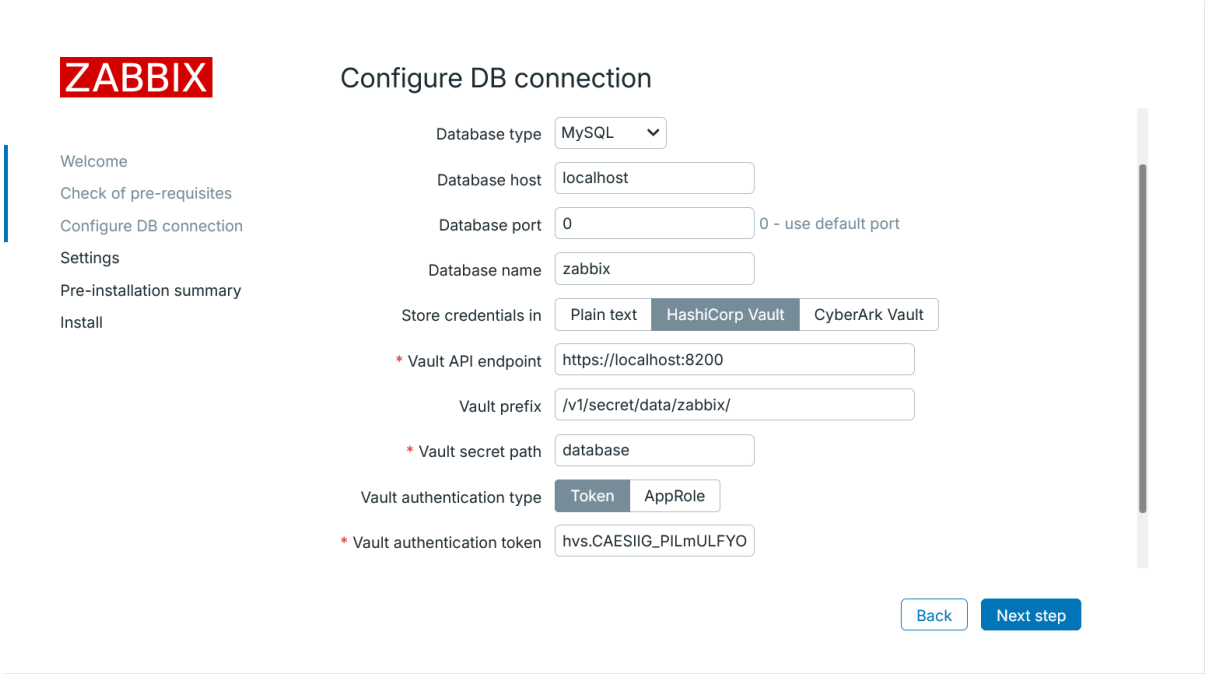
Frontend

Frontend Zabbix można skonfigurować tak, aby pobierał dane uwierzytelniające do bazy danych z sejfu podczas instalacji frontendu lub przez zaktualizowanie pliku konfiguracyjnego frontendu (`zabbix.conf.php`).

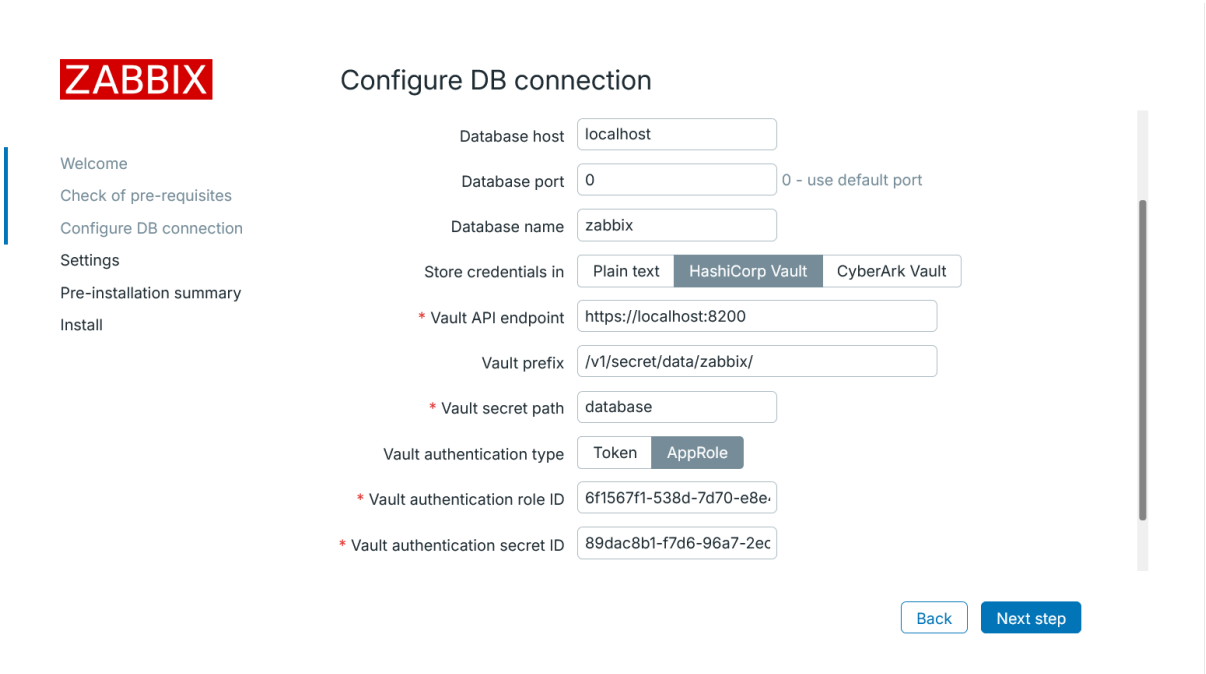
Attention:
Jeśli dane uwierzytelniające do sejfu zmieniły się od czasu poprzedniej instalacji frontendu, zaktualizuj istniejącą konfigurację.

Podczas **instalacji frontendu** parametry połączenia należy określić w kroku *Configure DB Connection*.

Przykład konfiguracji z uwierzytelnianiem za pomocą tokenu:



Przykład konfiguracji z uwierzytelnianiem AppRole:



Parameter	Mandatory	Default value	Description
Store credentials in	no		Wybierz **HashiCorp Vault**.
Vault API endpoint	yes	https://localhost:8200	Określ URL połączenia z sejfem w formacie `scheme://host:port`
Vault prefix	no	/v1/secret/data/	Podaj niestandardowy prefiks ścieżki lub zapytania sejfu. Jeśli nie zostanie określony, zostanie użyta wartość domyślna. Przykład: `/v1/secret/data/zabbix/`
Vault secret path	yes		Ścieżka do sekretu, z którego dane uwierzytelniające do bazy danych będą pobierane za pomocą kluczy „password” i „username”. Przykład: `database`
Vault authentication type	no		Wybierz typ uwierzytelniania sejfu: **Token** - użyj uwierzytelniania za pomocą tokenu; **AppRole** - użyj metody uwierzytelniania AppRole.

Parameter	Mandatory	Default value	Description
Vault authentication token	yes		Podaj token uwierzytelniania zapewniający dostęp tylko do odczytu do ścieżki sekretu. Informacje o tworzeniu tokenów i zasad sejfu znajdziesz w dokumentacji HashiCorp. To pole jest dostępne tylko wtedy, gdy jako typ uwierzytelniania sejfu wybrano *Token*.
Vault authentication role ID	yes		Podaj identyfikator roli AppRole. Informacje o AppRole znajdziesz w dokumentacji HashiCorp. To pole jest dostępne tylko wtedy, gdy jako typ uwierzytelniania sejfu wybrano *AppRole*.
Vault authentication secret ID	yes		Podaj identyfikator sekretu wydany dla AppRole. Informacje o AppRole znajdziesz w dokumentacji HashiCorp. To pole jest dostępne tylko wtedy, gdy jako typ uwierzytelniania sejfu wybrano *AppRole*.

Pobieranie wartości makr użytkownika

Aby używać HashiCorp Vault do przechowywania wartości makr użytkownika *Vault secret*, upewnij się, że:

- Zabbix server/proxy jest skonfigurowany do pracy z HashiCorp Vault.
- Parametr *Vault provider* w *Administration > General > Other* jest ustawiony na "HashiCorp Vault" (domyślnie).



Note:
Zabbix server (oraz Zabbix proxy, jeśli jest skonfigurowany) wymaga dostępu do wartości makr *Vault secret* z vault. Zabbix frontend nie potrzebuje takiego dostępu.

Wartość makra powinna zawierać ścieżkę referencyjną (w formacie `path:key`, na przykład `macros:password`). Token uwierzytelniający określony podczas konfiguracji Zabbix server/proxy (za pomocą parametru `VaultToken`) musi zapewniać dostęp tylko do odczytu do tej ścieżki.

Szczegółowe informacje o przetwarzaniu wartości makr przez Zabbix znajdziesz w sekcji *Vault secret macros*.

Składnia ścieżki

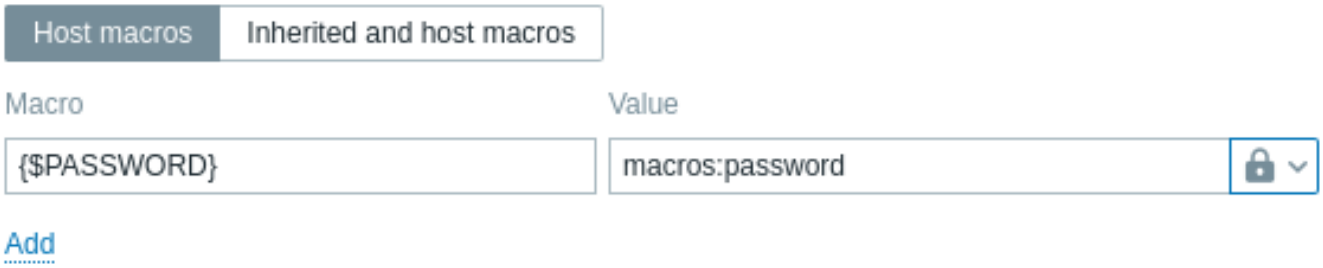
Symbole ukośnika ("/") i dwukropka (":") są zarezerwowane.

Ukośnik może być używany wyłącznie do oddzielenia punktu montowania od ścieżki (np. *secret/zabbix*, gdzie punktem montowania jest "secret", a ścieżką "zabbix"). W przypadku makr Vault dwukropek może być używany wyłącznie do oddzielenia ścieżki/zapytania od klucza.

Możliwe jest zakodowanie znaków ukośnika i dwukropka w formacie URL, jeśli istnieje potrzeba utworzenia punktu montowania o nazwie zawierającej ukośnik (np. *foo/bar/zabbix*, gdzie punktem montowania jest "foo/bar", a ścieżką "zabbix", można zakodować jako "foo%2Fbar/zabbix") oraz jeśli nazwa punktu montowania lub ścieżka muszą zawierać dwukropek.

Przykład

1. W Zabbix dodaj makro użytkownika {$PASSWORD} typu "Vault secret" i z wartością `macros:password`



2. Uruchom następujące polecenia CLI, aby utworzyć wymagany sekret w vault:

```
#### Włącz punkt montowania "secret/", jeśli nie jest jeszcze włączony; pamiętaj, że należy użyć "kv-v2".
vault secrets enable -path=secret/ kv-v2

#### Dodaj nowy sekret z kluczem "password" w punkcie montowania "secret/" i ścieżce "zabbix/macros".
vault kv put -mount=secret zabbix/macros password=<password>

#### Sprawdź, czy sekret został pomyślnie dodany.
vault kv get secret/zabbix/macros

#### Na koniec przetestuj za pomocą Curl; pamiętaj, że "data" należy ręcznie dodać po punkcie montowania,
curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/macros
```

3. W rezultacie Zabbix rozwiąże makro {$PASSWORD} do wartości: <password>

Aktualizowanie istniejącej konfiguracji

Aby zaktualizować istniejącą konfigurację pobierania sekretów z HashiCorp Vault:

1. Zaktualizuj parametry pliku konfiguracyjnego serwera Zabbix lub proxy zgodnie z opisem w sekcji *Dane uwierzytelniające bazy danych*.
2. Zaktualizuj parametry pliku konfiguracyjnego frontend Zabbix:

Korzystanie z tokenu Vault:

```
$DB['VAULT']                    = 'HashiCorp';
$DB['VAULT_URL']                = 'https://localhost:8200';
$DB['VAULT_DB_PATH']            = 'database';
$DB['VAULT_TOKEN']              = 'hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTYONFN
$DB['VAULT_PREFIX']             = '/v1/secret/data/zabbix/';
```

Lub korzystanie z metody Vault AppRole:

```
$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. W razie potrzeby skonfiguruj makra użytkownika zgodnie z opisem w sekcji *Wartości makr użytkownika*.

Aby zaktualizować istniejącą konfigurację pobierania sekretów z CyberArk Vault, zobacz sekcję *Konfiguracja CyberArk*.

Rozwiązywanie problemów

1. W pliku dziennika serwera/proxy Zabbix:

```
237839:20260729:130916.425 Vault token renew started to fail: cannot login into HashiCorp vault with AppRo
```

Prawdopodobna przyczyna: HashiCorp Vault został uruchomiony ponownie, ale nie został odblokowany (wymagane w konfiguracji domyślnej, niewymagane, jeśli skonfigurowano Auto Unseal).

2. W pliku dziennika serwera/proxy Zabbix:

```
235545:20260729:125617.944 Vault token renew started to fail: cannot login into HashiCorp vault with AppRo
```

lub

```
235545:20260729:125718.537 Vault token renew failed: cannot login into HashiCorp vault with AppRole method
```

Prawdopodobna przyczyna: VaultAppSecretID wygasł lub osiągnął limit użyć w HashiCorp vault.

Zaufanie do certyfikatu TLS w Google Chrome

Przegląd

Ta strona zawiera kroki konfiguracji Zabbix oraz przykłady konfiguracji dotyczące używania Google Chrome z frontendem Zabbix lub punktami końcowymi zabbix-web-service zabezpieczonymi certyfikatem samopodpisanym albo certyfikatem wydanym przez prywatne centrum certyfikacji.

Instrukcje te zakładają, że docelowy serwer WWW jest już skonfigurowany do pracy z HTTPS. Informacje o konfiguracji TLS dla frontend Zabbix można znaleźć w sekcji Bezpieczne połączenie z frontendem.

Konfiguracja

Google Chrome w systemie Linux używa bazy certyfikatów NSS dla certyfikatów zaufanych, osobnej dla każdego użytkownika. Aby sprawić, by Chrome ufał certyfikatowi samopodpisanemu lub prywatnemu urzędowi certyfikacji, dodaj wymagane certyfikaty do bazy NSS konta użytkownika, które uruchamia Chrome.

1. Zainstaluj wymagane pakiety.

Dla Debian/Ubuntu:

```
sudo apt install ca-certificates libnss3-tools
```

Dla systemów opartych na RHEL:

```
sudo dnf install ca-certificates nss-tools
```

Lub:

```
sudo yum install ca-certificates nss-tools
```

2. Przygotuj pliki certyfikatów.

Dla każdego wymaganego certyfikatu: utwórz plik `.crt` oparty na PEM dla certyfikatu samopodpisanego albo utwórz oddzielne pliki `.crt` oparte na PEM dla certyfikatu głównego urzędu certyfikacji oraz, jeśli to konieczne, dla każdego certyfikatu pośredniego.

3. Utwórz katalog bazy NSS.

Użyj katalogu domowego użytkownika, który uruchamia Chrome.

Na przykład:

```
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
```

Jeśli Chrome w systemie docelowym używa innej lokalizacji bazy NSS, użyj zamiast tego tego katalogu.

Upewnij się, że pliki certyfikatów są czytelne dla użytkownika, który uruchamia Google Chrome. W razie potrzeby skopiuj je do lokalizacji dostępnej dla tego użytkownika.

4. Zaimportuj certyfikaty.

Jeśli używasz certyfikatu samopodpisanego:

```
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
```

Jeśli używasz prywatnego urzędu certyfikacji, zaimportuj do bazy certyfikat głównego urzędu certyfikacji:

```
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
```

Jeśli używane są jakiekolwiek certyfikaty pośrednie, zaimportuj każdy z nich osobno:

```
sudo -u zabbix certutil -d sql:/var/lib/zabbix/.pki/nssdb \
  -A -t ",," \
  -n "Zabbix intermediate certificate" \
  -i /path/to/intermediate-ca.crt
```

5. Zweryfikuj bazę certyfikatów.

```
sudo -u zabbix certutil -d sql:/var/lib/zabbix/.pki/nssdb -L
```

6. Przetestuj Chrome.

Google Chrome używa bazy certyfikatów konta użytkownika, który go uruchamia. Jeśli konto użytkownika nie ma zapisywalnego katalogu domowego, skonfiguruj zapisywalny katalog domowy oraz katalogi XDG przed uruchomieniem testu.

Przykład (zastąp `google-chrome` przez `chromium`, jeśli używany jest 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
```

Jeśli konfiguracja jest poprawna, polecenie zwróci kod HTML żądanej strony.

Jeśli Chrome nie ufa certyfikatowi, polecenie zwróci stronę błędu certyfikatu zawierającą komunikaty takie jak `NET::ERR_CERT_AUTHORITY_I`

Rozwiązywanie problemów

Błąd	Możliwa przyczyna
`NET::ERR_CERT_AUTHORITY_INVALID`	Certyfikat został zaimportowany do niewłaściwego profilu użytkownika, brakuje certyfikatu urzędu certyfikacji głównego lub brakuje certyfikatu pośredniego.
`NET::ERR_CERT_COMMON_NAME_INVALID`	Certyfikat nie pasuje do nazwy hosta użytej w adresie URL przeglądarki.
`NET::ERR_CERT_DATE_INVALID`	Certyfikat wygasł, nie jest jeszcze ważny lub zegar systemowy jest nieprawidłowy.
Chrome nadal wyświetla ostrzeżenie o certyfikacie po imporcie	Chrome używa innej bazy danych NSS dla bieżącego użytkownika albo flagi zaufania certyfikatu są nieprawidłowe.

Najlepsze praktyki dotyczące bezpieczeństwa

Przegląd Ta sekcja zawiera najlepsze praktyki dotyczące bezpiecznej konfiguracji Zabbix.

Praktyki opisane w tej sekcji nie są wymagane do działania Zabbix, ale są zalecane w celu zwiększenia bezpieczeństwa systemu.

Aby poznać najlepsze praktyki dotyczące zabezpieczania aplikacji mobilnej Zabbix i adaptera Zabbix bridge, zobacz Bezpieczny dostęp mobilny.

Kodowanie UTF-8 UTF-8 jest jedynym kodowaniem obsługiwanym przez Zabbix. Wiadomo, że działa ono bez żadnych luk bezpieczeństwa. Użytkownicy powinni mieć świadomość, że w przypadku używania niektórych innych kodowań występują znane problemy bezpieczeństwa.

Ścieżki instalatora Windows Podczas korzystania z instalatorów Windows zaleca się używanie domyślnych ścieżek udostępnianych przez instalator. Używanie niestandardowych ścieżek bez odpowiednich uprawnień może zagrozić bezpieczeństwu instalacji.

Makra w globalnych skryptach zdefiniowanych przez użytkownika Aby zwiększyć bezpieczeństwo, zaleca się używanie funkcji makr zamiast zwykłych makr w globalnych skryptach zdefiniowanych przez użytkownika, ponieważ makra nie są automatycznie escape'owane.

Ostrzeżenia bezpieczeństwa Zabbix i baza danych CVE Zobacz *Ostrzeżenia bezpieczeństwa Zabbix i baza danych CVE*.

Szablony wiadomości e-mail w formacie HTML Podczas tworzenia lub edytowania szablonów wiadomości używanych w wiadomościach e-mail w formacie HTML zawsze otaczaj każde makro funkcją `htmlencode()`. Na przykład:

```
<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
```

Użycie `htmlencode()` gwarantuje, że wszystkie znaki HTML w wartościach makr zostaną zamienione na encje i zapobiega wstrzykiwaniu kodu HTML do powiadomień (na przykład wstawieniu przez atakującego złośliwego lub phishingowego odnośnika do powiadomienia).

Uwaga: domyślne wiadomości e-mail w formacie HTML dostarczane przez Zabbix już stosują `htmlencode()` do makr. Ta wskazówka dotyczy edytowania istniejących szablonów i tworzenia nowych — przed użyciem szablonu do wysyłania wiadomości e-mail w formacie HTML sprawdź, czy makra są kodowane.

Interfejs webowy

Przegląd Aby zapewnić łatwy dostęp do Zabbix z dowolnego miejsca i dowolnej platformy, udostępniono interfejs oparty na sieci Web.

> **Note:**
> W przypadku korzystania z więcej niż jednej instancji frontend upewnij się, że ustawienia regionalne i biblioteki (LDAP, SAML itp.) są zainstalowane i skonfigurowane identycznie dla wszystkich frontendów.

Pomoc frontendu Link do pomocy ? znajduje się w sekcjach i formularzach frontendu Zabbix w prawym górnym rogu i prowadzi bezpośrednio do odpowiednich części dokumentacji.

W przypadku pól formularza z ikoną pomocy ● umieszczenie wskaźnika myszy nad ikoną wyświetla odpowiednią podpowiedź.

Tryb kiosku W przypadku sekcji frontend z przyciskiem *Tryb kiosku* (znajdującym się w prawym górnym rogu) kliknij ten przycisk, aby przełączyć sekcję w tryb kiosku. W trybie kiosku wyświetlana jest tylko zawartość sekcji.

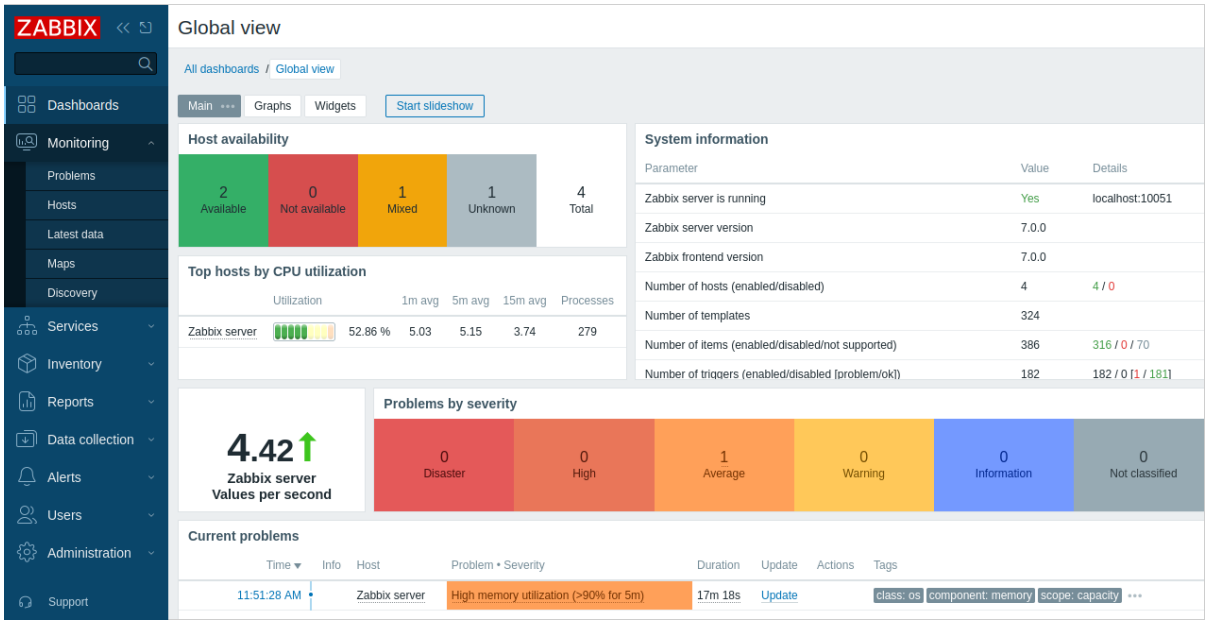
Aby wyjść z trybu kiosku, najedź kursorem na zawartość i kliknij przycisk *Widok normalny* , który pojawi się w prawym górnym rogu.

Menu

Przegląd

Pionowe menu na pasku bocznym zapewnia dostęp do różnych sekcji frontendu Zabbix.

Domyślny motyw posiada ciemnoniebieskie menu.

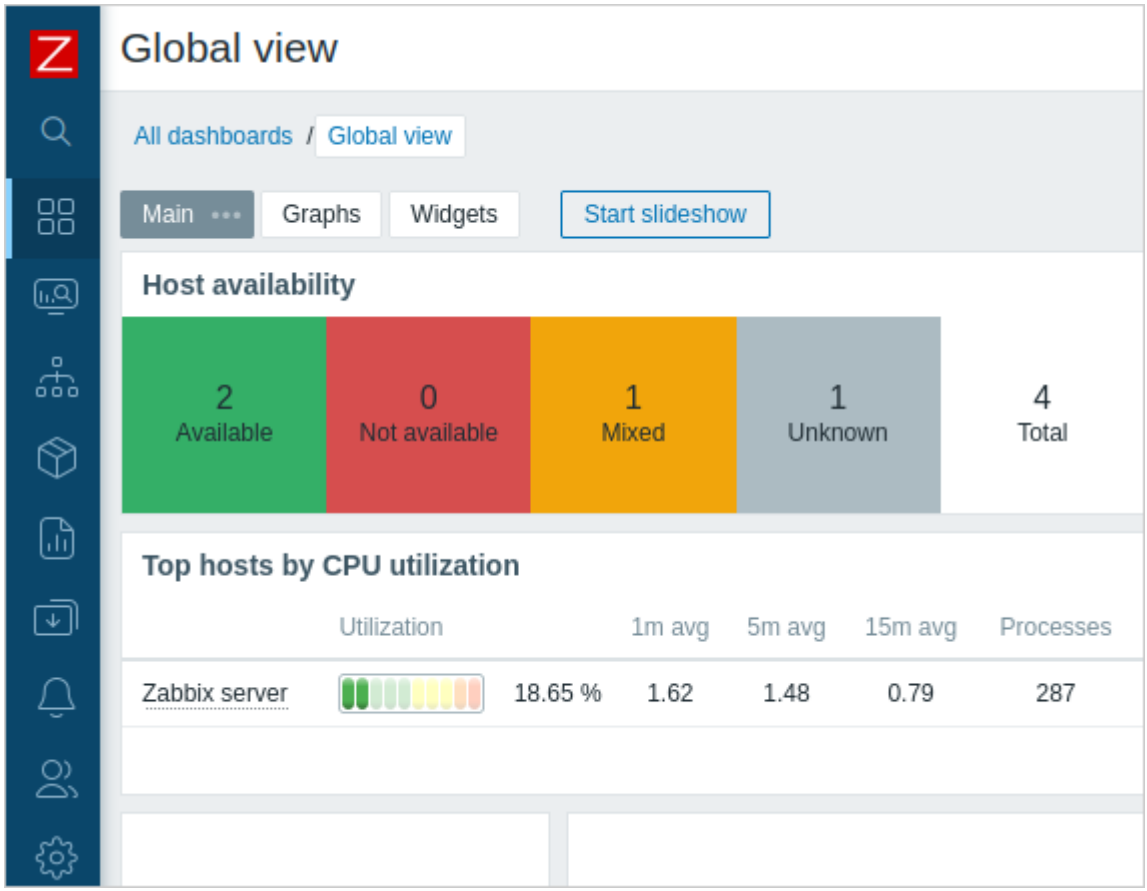


Praca z menu

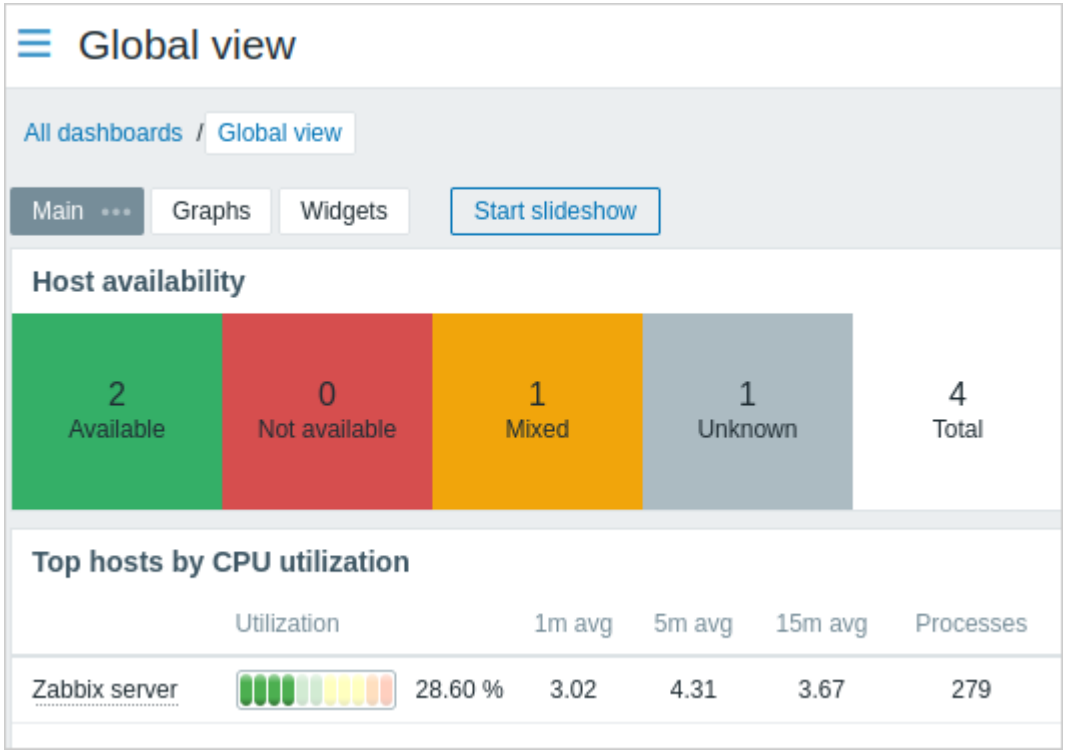
Pole global search znajduje się pod logo Zabbix.

Menu można zwinąć lub całkowicie ukryć:

- Aby zwinąć menu, kliknij  obok logo Zabbix. W zwiniętym menu widoczne są tylko ikony.



- Aby ukryć menu, kliknij  obok logo Zabbix. W ukrytym menu wszystko jest niewidoczne.



Zwinięte menu

Gdy menu jest zwinięte i wyświetlane wyłącznie jako ikony, pełne menu pojawi się ponownie, gdy tylko umieścisz nad nim kursor myszy. Pamiętaj, że pojawi się ono nad zawartością strony; aby przesunąć zawartość strony w prawo, musisz kliknąć przycisk rozwijania. Jeśli ponownie umieścisz kursor myszy poza pełnym menu, menu zostanie ponownie zwinięte po dwóch sekundach.

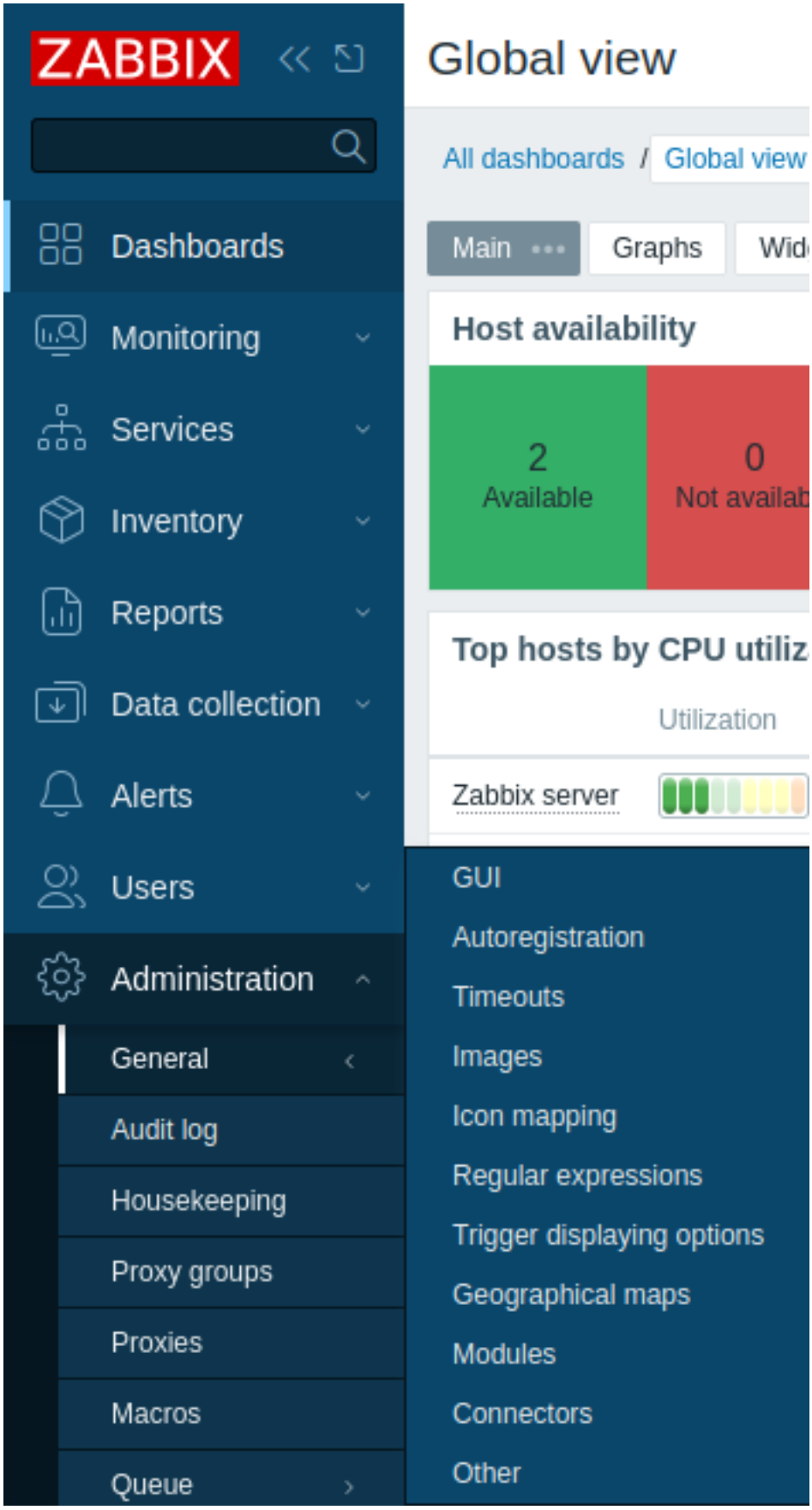
Możesz również ponownie w pełni wyświetlić zwinięte menu, naciskając klawisz Tab. Wielokrotne naciskanie klawisza Tab umożliwia przechodzenie do kolejnych elementów menu.

Ukryte menu

Nawet gdy menu jest całkowicie ukryte, pełne menu jest dostępne po jednym kliknięciu myszą, poprzez kliknięcie ikony burgera  . Należy pamiętać, że pojawia się ono ponownie nad zawartością strony; aby przesunąć zawartość strony w prawo, trzeba odkryć menu, klikając przycisk pokaż pasek boczny.

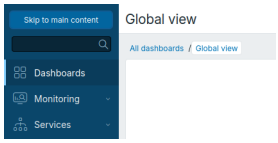
Poziomy menu

W menu są maksymalnie trzy poziomy.



Dostępność: Pomijanie nawigacji

Link *Przejdź do głównej treści* umożliwia użytkownikom korzystającym z klawiatury i czytników ekranu pominięcie powtarzalnej nawigacji w panelu bocznym i przejście bezpośrednio do głównej treści.



Pozostaje ukryty podczas normalnego korzystania i pojawia się tylko po uzyskaniu fokusu, na przykład po naciśnięciu klawisza Tab po załadowaniu strony lub podczas korzystania z czytnika ekranu.

Menu kontekstowe

Oprócz menu głównego Zabbix udostępnia menu kontekstowe dla host, pozycja oraz zdarzenie, zapewniające szybki dostęp do często używanych elementów, takich jak najnowsze wartości, prosty wykres, formularz konfiguracji, powiązane skrypty lub linki zewnętrzne.

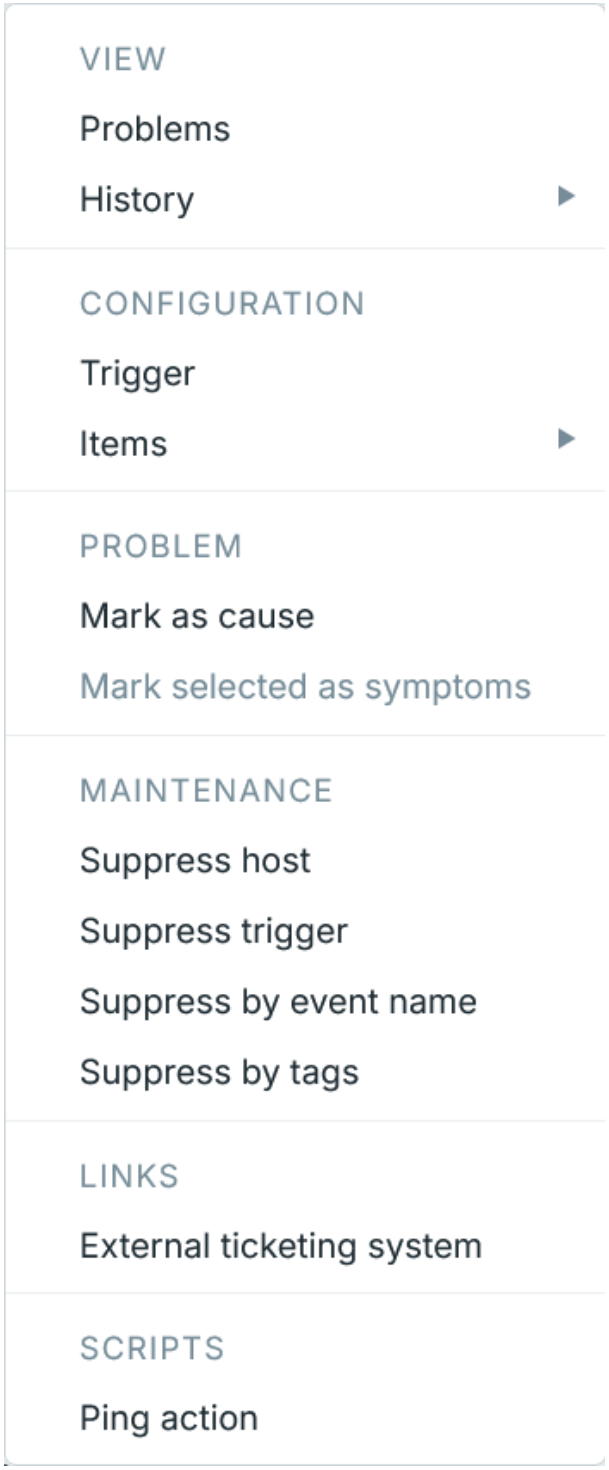
Do menu kontekstowych można uzyskać dostęp, klikając nazwę hosta, pozycji lub problemu/wyzwalacza w obsługiwanych lokalizacjach.

Menu zdarzeń

Przegląd

Menu zdarzenia zawiera skróty do działań lub sekcji frontend, które są często wymagane w przypadku zdarzenia.

Menu zdarzenia można otworzyć, klikając nazwę zdarzenia.



Zawartość

Menu kontekstowe zdarzenia zawiera kilka sekcji: *Widok*, *Działania*, *Konfiguracja*, *Problem*, *Konserwacja*, *Łącza* i *Skrypty*. W przypadku nieskonfigurowanych encji łącza są wyłączone i wyświetlane na szaro. Sekcje *Skrypty* i *Łącza* są wyświetlane, jeśli odpowiadające im encje są skonfigurowane.

Sekcja *Widok* zawiera łącza do:

- **Problemy** - otwiera listę nierozwiązanych problemów powiązanego wyzwalacza;
- **Historia** - prowadzi do wykresu *Najnowsze dane* / historii pozycji dla powiązanej pozycji lub pozycji. Jeśli wyzwalacz używa kilku pozycji, łącza będą dostępne dla każdej z nich.

Sekcja *Działania* jest dostępna tylko w widżetach *Przegląd wyzwalaczy*. Zawiera łącze do:

- **Aktualizuj problem** - otwiera ekran aktualizacji problemu.

Sekcja *Konfiguracja* zawiera łącza do konfiguracji:

- **Wyzwalacza**, który spowodował problem;
- **Pozycji** używanych w wyrażeniu wyzwalacza.

Note:
Należy pamiętać, że sekcja konfiguracji jest dostępna tylko dla użytkowników z rolą Admin i Super admin.

Sekcja *Problem* zawiera opcje:

- **Oznacz jako przyczynę** - oznacza problem jako przyczynę;
- **Oznacz wybrane jako objawy** - oznacza wybrane problemy jako objawy tego problemu.

Sekcja *Konserwacja* zawiera opcje uruchomienia konserwacji ad hoc:

- **Wstrzymaj host** - rozpoczyna okres konserwacji dla hosta;
- **Wstrzymaj wyzwalacz** - rozpoczyna okres konserwacji dla wyzwalacza;
- **Wstrzymaj według nazwy zdarzenia** - rozpoczyna okres konserwacji dla nazwy zdarzenia;
- **Wstrzymaj według tagów** - rozpoczyna okres konserwacji dla zdarzeń z odpowiednimi tagami.

Po wybraniu odpowiedniej opcji otworzy się formularz konfiguracji okresu konserwacji.

Jednorazowy okres konserwacji trwający godzinę (konfigurowalny w ustawieniach użytkownika i jego profilu), rozpoczynający się od teraz, zostanie wstępnie wybrany. Odpowiedni host, wyzwalacz, nazwa zdarzenia lub tagi również zostaną wstępnie wybrane. Nazwa konserwacji zostanie wstępnie wypełniona w formacie `Ad-hoc: <event name>`.

Sekcja *Łącza* zawiera łącza umożliwiające:

- dostęp do skonfigurowanego URL wyzwalacza;
- dostęp do niestandardowych łączy skonfigurowanych w sekcji Skrypty globalne (z zakresem „Ręczne działanie na zdarzeniu” i typem „URL”);
- dostęp do skonfigurowanego zewnętrznego zgłoszenia dotyczącego problemu (zobacz opcję *Uwzględnij wpis w menu zdarzenia* podczas konfigurowania webhooków).

Sekcja *Skrypty* zawiera łącza umożliwiające wykonanie globalnego skryptu (z zakresem *Ręczne działanie na zdarzeniu*). Ta funkcja może być przydatna do uruchamiania skryptów używanych do zarządzania zgłoszeniami dotyczącymi problemów w systemach zewnętrznych.

Obsługiwane lokalizacje

Menu kontekstowe zdarzenia jest dostępne po kliknięciu nazwy problemu lub zdarzenia w różnych sekcjach frontend, na przykład:

- Pulpity widżety, takie jak widżet *Problems*, widżet *Trigger overview* itd.
- *Monitoring* > *Problems*
- *Monitoring* > *Problems* > Szczegóły zdarzenia
- *Reports* > *Top 100 triggers* (globalne skrypty i dostęp do zewnętrznego zgłoszenia nie są obsługiwane w tej lokalizacji)

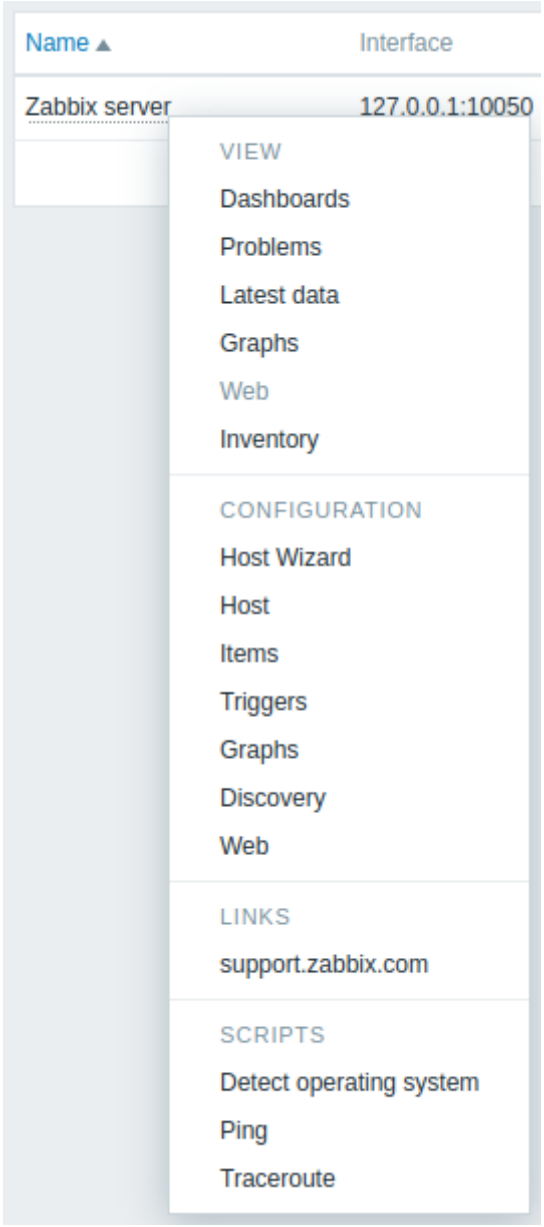
Menu hosta

Omówienie

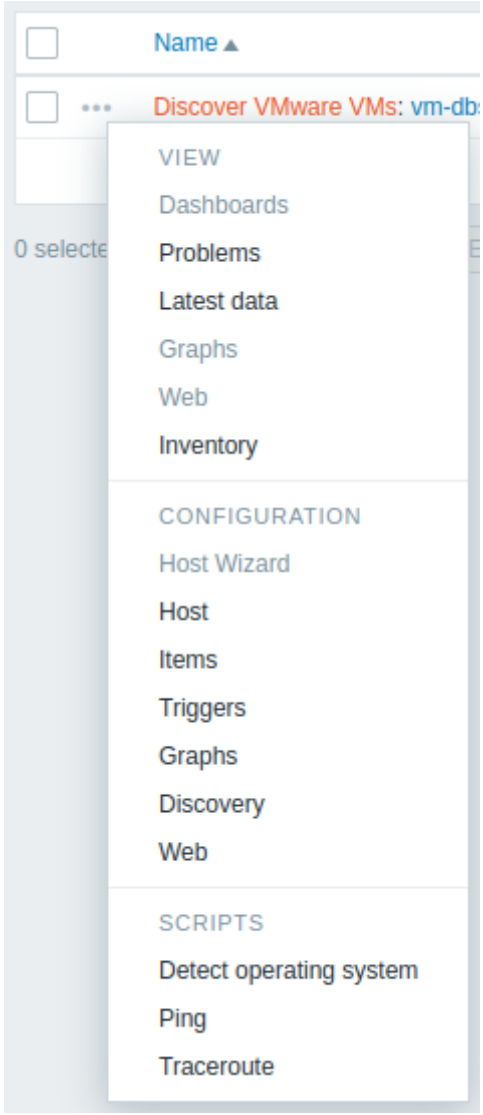
Menu hosta zawiera skróty do akcji lub sekcji frontend, które są często potrzebne dla hosta.

Menu hosta można otworzyć, klikając nazwę hosta lub ikonę trzech kropek, w zależności od sekcji frontend, na przykład:

Monitoring > *Latest data*



Data collection > *Hosts*



Zawartość

Menu kontekstowe hosta zawiera cztery sekcje: *Widok*, *Konfiguracja*, *Łącza* i *Skrypty*. W przypadku nie skonfigurowanych encji łącza są wyłączone i wyświetlane na szaro. Sekcje *Skrypty* i *Łącza* są wyświetlane, jeśli ich encje są skonfigurowane.

Sekcja *Widok* zawiera łącza do:

- **Pulpity nawigacyjne** - otwiera widżety i wykresy.
- **Problemy** - otwiera sekcję *Problemy* z listą nierozwiązanych problemów powiązanych z wyzwalaczem.
- **Najnowsze dane** - otwiera sekcję *Najnowsze dane* z listą wszystkich najnowszych danych bieżącego hosta.
- **Wykresy** - otwiera proste wykresy bieżącego hosta.
- **Web** - otwiera łącze do skonfigurowanych scenariuszy web.
- **Inwentaryzacja** - otwiera łącze do inwentaryzacji bieżącego hosta.

Sekcja *Konfiguracja* zawiera łącza do:

- **Kreator hosta** - otwiera Kreatora hosta dla bieżącego hosta (wyłączone w przypadku hostów wykrytych).
- **Host** - formularz konfiguracji bieżącego hosta.
- **Pozycje** - lista pozycji bieżącego hosta.
- **Wyzwalacze** - lista wyzwalaczy bieżącego hosta.
- **Wykresy** - proste wykresy bieżącego hosta.
- **Wykrywanie** - lista reguł wykrywania niskiego poziomu bieżącego hosta.
- **Web** - lista scenariuszy web bieżącego hosta.

Note:
Należy pamiętać, że sekcja konfiguracji jest dostępna tylko dla użytkowników typu Admin i Super admin.

Sekcja *Łącza* zawiera łącza do:

- skonfigurowanego adresu URL wyzwalacza.
- niestandardowych łączy skonfigurowanych w sekcji Skrypty globalne (z zakresem *Ręczna akcja hosta* i typem „URL").

Sekcja *Skrypty* umożliwia wykonywanie skryptów globalnych skonfigurowanych dla bieżącego hosta. Aby były dostępne w menu hosta, skrypty te muszą mieć zdefiniowany zakres *Ręczna akcja hosta*.

Obsługiwane lokalizacje

Menu hosta jest dostępne po kliknięciu nazwy hosta w różnych sekcjach frontend, na przykład:

- Pulpity widżety, takie jak *Problems*, *Top items*, *Trigger overview* itd.
- *Monitoring* > *Problems*
- *Monitoring* > *Problems* > Szczegóły zdarzenia
- *Monitoring* > *Hosts*
- *Monitoring* > *Hosts* > *Web Monitoring*
- *Monitoring* > *Latest data*
- *Monitoring* > *Maps*
- *Inventory* > *Hosts*
- *Reports* > *Top 100 triggers*

Menu hosta jest dostępne po kliknięciu ikony trzech kropek w sekcji Data collection > Hosts.

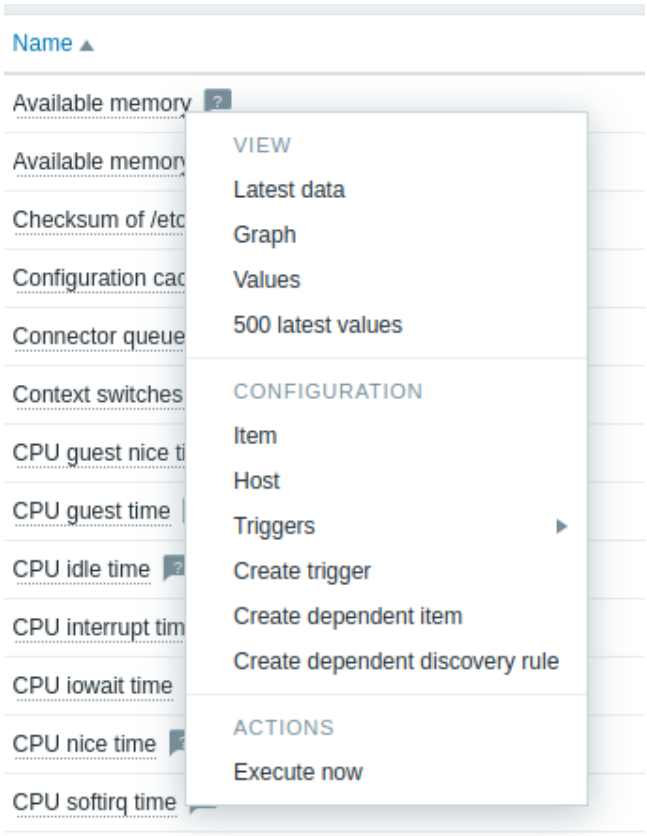
Menu pozycji

Omówienie

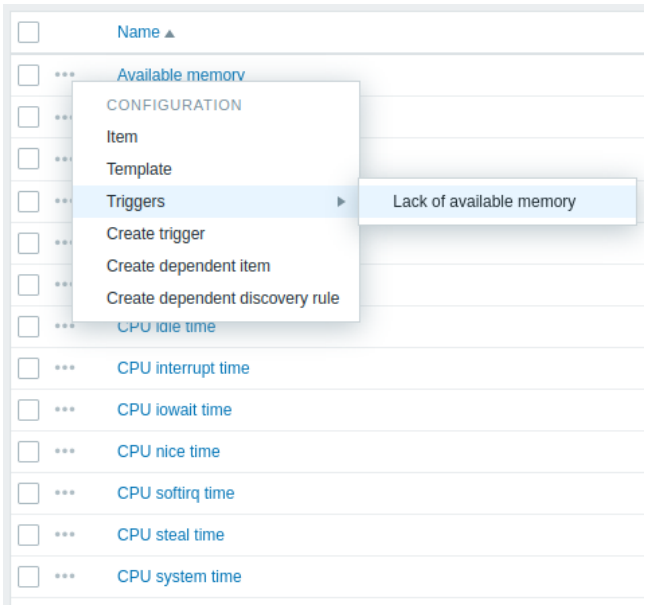
Menu pozycji zawiera skróty do akcji lub sekcji frontend, które są często potrzebne dla pozycji.

Menu pozycji można otworzyć, klikając nazwę pozycji lub ikonę z trzema kropkami, w zależności od sekcji frontend, na przykład:

Monitoring > *Latest data*



Data Collection > *Templates* > *Items*



Zawartość

Menu pozycji zawiera trzy sekcje: *Widok*, *Konfiguracja* i *Działania*.

Sekcja *Widok* zawiera następujące opcje:

- **Najnowsze dane** - otwiera sekcję *Najnowsze dane* przefiltrowaną według bieżącego hosta i pozycji;
- **Wykres** - otwiera prosty wykres bieżącej pozycji;
- **Wartości** - otwiera listę wszystkich wartości odebranych dla bieżącej pozycji w ciągu ostatnich 60 minut;
- **500 najnowszych wartości** - otwiera listę 500 najnowszych wartości bieżącej pozycji.

Sekcja *Konfiguracja* (dostępna tylko dla użytkowników typu *Admin* i *Super admin*) zawiera następujące opcje:

- **Pozycja** - otwiera formularz konfiguracji pozycji bieżącej pozycji;
- **Szablon** - otwiera formularz konfiguracji szablonu, do którego należy pozycja (dostępne tylko wtedy, gdy menu pozycji jest otwierane z sekcji *Zbieranie danych* > *Szablony* > *Pozycje*);
- **Host** - otwiera formularz konfiguracji hosta, do którego należy pozycja;
- **Wyzwalacze** - po najechaniu kursorem otwiera listę wyzwalaczy pozycji, jeśli takie istnieją; kliknięcie wyzwalacza otwiera jego formularz konfiguracji wyzwalacza;
- **Utwórz wyzwalacz** - otwiera formularz konfiguracji wyzwalacza w celu utworzenia wyzwalacza dla tej pozycji;
- **Utwórz pozycję zależną** - otwiera formularz konfiguracji pozycji w celu utworzenia pozycji zależnej, z bieżącą pozycją jako pozycją nadrzędną;
- **Utwórz zależną regułę wykrywania** - otwiera formularz konfiguracji reguły wykrywania w celu utworzenia zależnej reguły wykrywania, z bieżącą pozycją jako pozycją nadrzędną.

Sekcja *Działania* zawiera następującą opcję:

- **Wykonaj teraz** - wykonuje sprawdzenie w celu natychmiastowego uzyskania nowej wartości pozycji.

Obsługiwane lokalizacje

Menu pozycji jest dostępne po kliknięciu nazwy pozycji w różnych sekcjach frontend, na przykład:

- *Monitoring* > *Latest data*
- *Data collection* > *Hosts* > *Items*
- *Data collection* > *Hosts* > *Discovery rules* > *Item prototypes*

Menu pozycji jest dostępne po kliknięciu wartości pozycji w widżecie pulpitu *Top items*.

Sekcje frontend

Struktura menu Menu frontend Zabbix ma następującą strukturę:

- Pulpity
- Monitorowanie
  - Problemy
  - Hosty
  - Najnowsze dane
  - Mapy
  - Wykrywanie
- APM
  - Ślady
  - Metryki
  - Dzienniki
- Usługi
  - Usługi
  - SLA
  - Raport SLA
- Inwentaryzacja
  - Przegląd
  - Hosty
- Raporty
  - Informacje o systemie
  - Raporty zaplanowane
  - Raport dostępności
  - 100 najważniejszych wyzwalaczy
  - Dziennik audytu
  - Dziennik działań

    - Powiadomienia
- Gromadzenie danych
    - Grupy szablonów
    - Grupy hostów
    - Szablony
    - Hosty
    - Konserwacja
    - Korelacja zdarzeń
    - Wykrywanie
- Alerty
    - Działania
        * Działania wyzwalaczy
        * Działania usług
        * Działania wykrywania
        * Działania automatycznej rejestracji
        * Działania wewnętrzne
    - Typy nośników
    - Skrypty
- Użytkownicy
    - Grupy użytkowników
    - Role użytkowników
    - Użytkownicy
    - Tokeny API
    - Uwierzytelnianie
    - Urządzenia
- Administracja
    - Ogólne
        * GUI
        * Automatyczna rejestracja
        * Limity czasu
        * Obrazy
        * Mapowanie ikon
        * Wyrażenia regularne
        * Opcje wyświetlania wyzwalaczy
        * Mapy geograficzne
        * Moduły
        * Łączniki
        * Inne
    - Dziennik audytu
    - Porządkowanie
    - Grupy proxy
    - Proxy
    - Makra
    - Kolejka
        * Przegląd kolejki
        * Przegląd kolejki według proxy
        * Szczegóły kolejki

Pulpity nawigacyjne

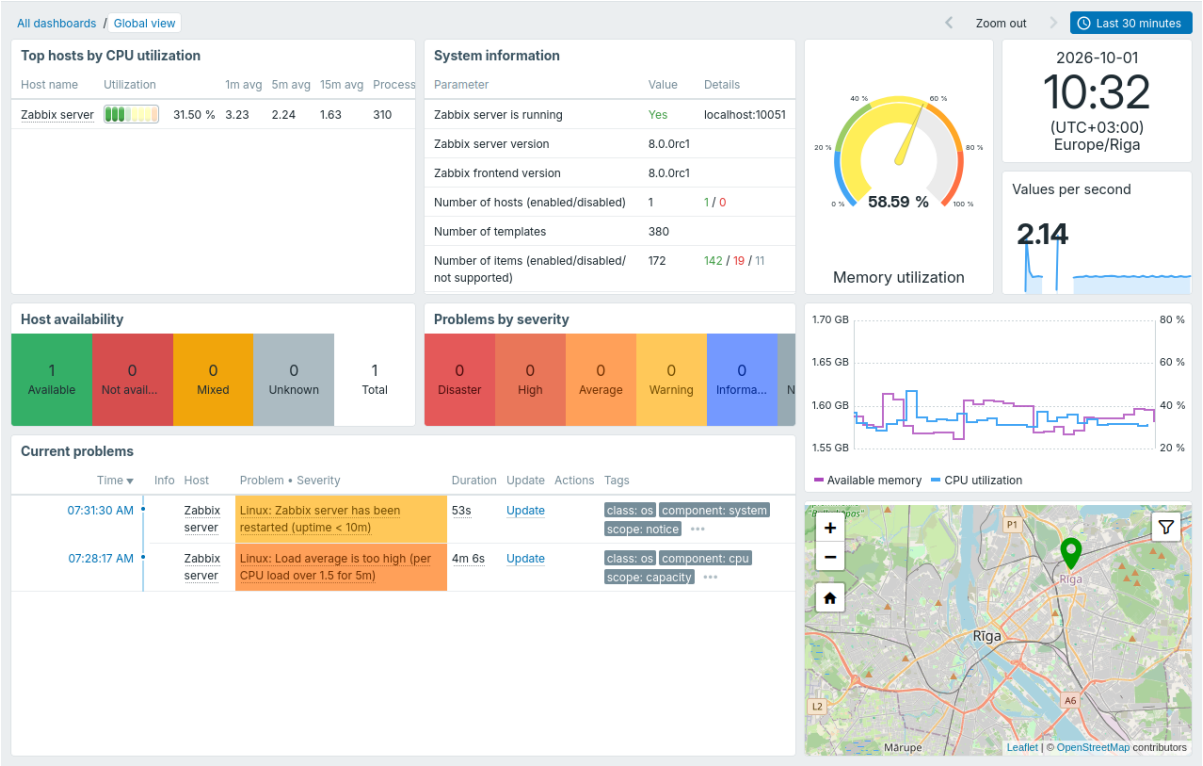
Pulpity nawigacyjne zapewniają przegląd monitorowanego środowiska. Możesz je tworzyć i dostosowywać do swoich potrzeb.

Elementami składowymi są widżety: wykres, mapa, lista ostatnich problemów i inne, z których każdy jest przeznaczony do wyświetlania określonego typu informacji.

Widżety można organizować na wielu stronach pulpitu nawigacyjnego. Takie strony można wyświetlać jako pokaz slajdów. Możesz także przejść do dowolnej strony, klikając jej kartę.

Możesz utworzyć kilka pulpitów nawigacyjnych, ale w danym momencie wyświetlany jest tylko jeden.

Zabbix zawiera kilka domyślnych pulpitów nawigacyjnych. Jednym z nich jest pulpit nawigacyjny *Widok globalny*, który pokazuje kondycję i stan monitorowanych hostów, w tym serwera Zabbix:

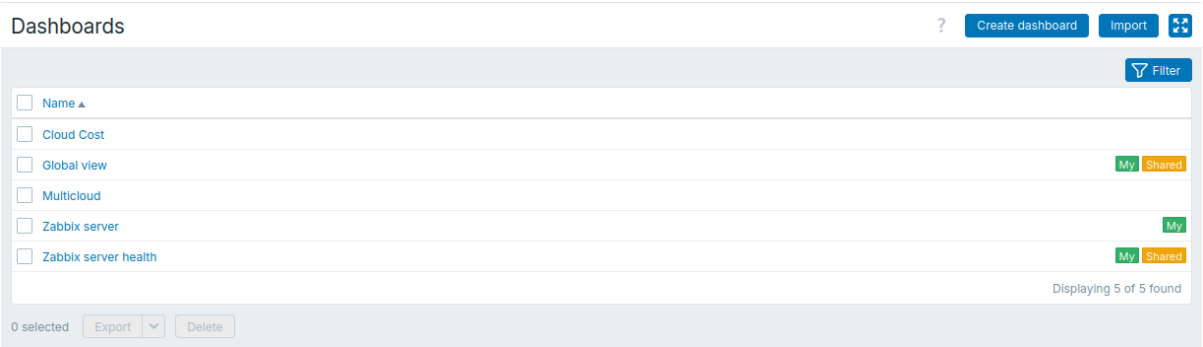


Więcej pulpitów nawigacyjnych znajdziesz w repozytorium pulpitów nawigacyjnych. Te pulpity nawigacyjne są przeznaczone do pracy z istniejącymi szablonami Zabbix i można je pobrać oraz zaimportować do środowiska Zabbix. Szczegółowy opis każdego pulpitu nawigacyjnego, w tym wymagania wstępne, znajduje się w pliku `README` danego pulpitu nawigacyjnego.

Lista pulpitów

Aby otworzyć listę pulpitów, wykonaj jedną z następujących czynności:

- Kliknij *Dashboards* w menu.
- Kliknij *All dashboards* podczas wyświetlania pulpitu, aby otworzyć listę pulpitów.



Dostęp do menu *Dashboards* zależy od roli użytkownika oraz uprawnienia *Dashboard* w sekcji *Access to UI elements*.

W tym miejscu możesz: - Tworzyć pulpity. - Usuwać pulpity. Aby usunąć pulpit, zaznacz jego pole wyboru i kliknij przycisk *Delete*. - Importować pulpity. - Eksportować pulpity.

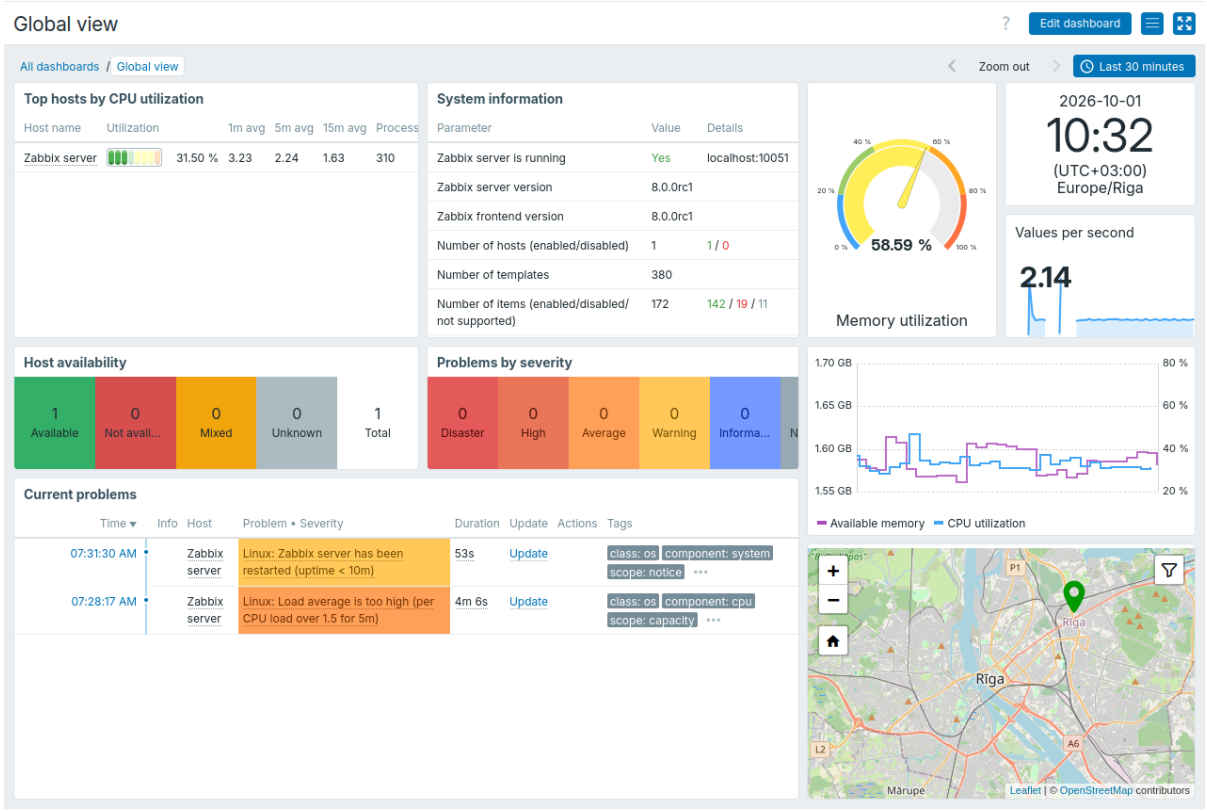
Pulpity mają znaczniki wskazujące ich właściciela i ustawienia udostępniania:

- *My* - pulpit należy do Ciebie i jest widoczny tylko dla Ciebie.
- *My* + *Shared* - pulpit należy do Ciebie, ale jest publiczny lub udostępniony określonym użytkownikom albo grupom użytkowników.
- Pulpity bez znaczników należą do innych użytkowników, ale są publiczne lub udostępnione Tobie.

Listę można filtrować według nazwy pulpitu i właściciela.

Wyświetlanie dashboardu

Kliknij nazwę dashboardu na liście dashboardów, aby wyświetlić dashboard:



Interfejs użytkownika Zabbix zapamiętuje ostatnio otwarty dashboard na potrzeby następnej sesji.

Note:
Dane widoczne na dashboardzie zależą od dostępu do hostów.

W przypadku widgetów wyświetlających wykresy selektor okresu znajdujący się nad widgetami ustawia zakres czasu. Wyświetla on bieżący zakres czasu jako etykietę. Kliknij etykietę, aby otworzyć lub zamknąć selektor.

Z tego miejsca możesz zarządzać dashboardami w następujący sposób:

- Edytuj dashboard.
- Wykonuj następujące działania z menu *Actions* dashboardu :
  - *Sharing* - edytuj ustawienia udostępniania dashboardu.
  - *Create new* - utwórz nowy dashboard.
  - *Clone* - utwórz nowy dashboard, klonując bieżący. Przed zapisaniem możesz edytować właściwości dashboardu.
  - *Delete* - usuń dashboard.
  - *Export* - wyeksportuj dashboard.
  - *Create new report* - utwórz raport cykliczny. Ta opcja jest wyłączona, jeśli nie masz uprawnień do zarządzania raportami cyklicznymi.
  - *View related reports* - wyświetl listę istniejących raportów dotyczących bieżącego dashboardu. Ta opcja jest wyłączona, jeśli nie ma powiązanych raportów lub nie masz uprawnień do zarządzania raportami cyklicznymi.
- Przełącz na tryb kiosku.
- Uruchom lub zatrzymaj pokaz slajdów dashboardu.

W trybie wyświetlania dashboardu dostępne są następujące opcje dotyczące widgetów:

- Edytuj widget.
- Wykonuj następujące działania z menu *Actions* widgetu :
  - *Copy* - skopiuj widget.
  - *Download image* - pobierz widget jako obraz PNG. Ta opcja jest dostępna dla widgetów obsługujących tę funkcję (zobacz strony poszczególnych widgetów).
  - *Refresh interval* - wybierz częstotliwość odświeżania zawartości widgetu.

W przypadku dashboardów z wieloma stronami w trybie wyświetlania dashboardu dostępne są następujące działania w menu *Actions* strony :

- *Copy* - skopiuj stronę.
- *Properties* - dostosuj właściwości strony, w tym jej nazwę i interwał pokazu slajdów.

Pokaz slajdów

Jeśli w właściwościach dashboardu zaznaczono pole wyboru *Uruchom pokaz slajdów*, pokaz slajdów rozpoczyna się automatycznie, gdy dashboard ma więcej niż jedną stronę. W przeciwnym razie pokaz slajdów można uruchomić, klikając przycisk *Uruchom pokaz slajdów*.



Strony zmieniają się zgodnie z interwałami określonymi we właściwościach dashboardu i poszczególnych stron. Jeśli dla pojedynczej strony zdefiniowano właściwość *Interwał pokazu slajdów*, stosowany jest interwał określony dla tej strony. W przeciwnym razie stosowany jest *Domyślny interwał pokazu slajdów* dashboardu.

Aby zatrzymać pokaz slajdów, kliknij *Zatrzymaj pokaz slajdów*.

W trybie kiosku użyj następujących przycisków:

- , aby uruchomić pokaz slajdów;
- , aby zatrzymać pokaz slajdów;
- , aby przejść do poprzedniej strony;
- , aby przejść do następnej strony.

Udostępnianie dashboardu

Dashboardy mogą być publiczne lub prywatne.

Publiczne dashboardy są widoczne dla wszystkich użytkowników. Prywatne dashboardy są widoczne tylko dla ich właścicieli oraz użytkowników lub grup użytkowników, którym je udostępniono.

Aby zmienić ustawienia udostępniania dashboardu, kliknij *Udostępnianie* w menu *Działania* dashboardu podczas wyświetlania dashboardu:

Parametr	Opis
Typ	Wybierz typ dashboardu: **Prywatny** - dashboard jest widoczny tylko dla wybranych grup użytkowników i użytkowników. **Publiczny** - dashboard jest widoczny dla wszystkich użytkowników.
Lista udostępnień dla grup użytkowników	Wybierz grupy użytkowników, które mogą uzyskać dostęp do dashboardu. Możesz przyznać dostęp tylko do odczytu lub do odczytu i zapisu.
Lista udostępnień dla użytkowników	Wybierz użytkowników, którzy mogą uzyskać dostęp do dashboardu. Możesz przyznać dostęp tylko do odczytu lub do odczytu i zapisu.

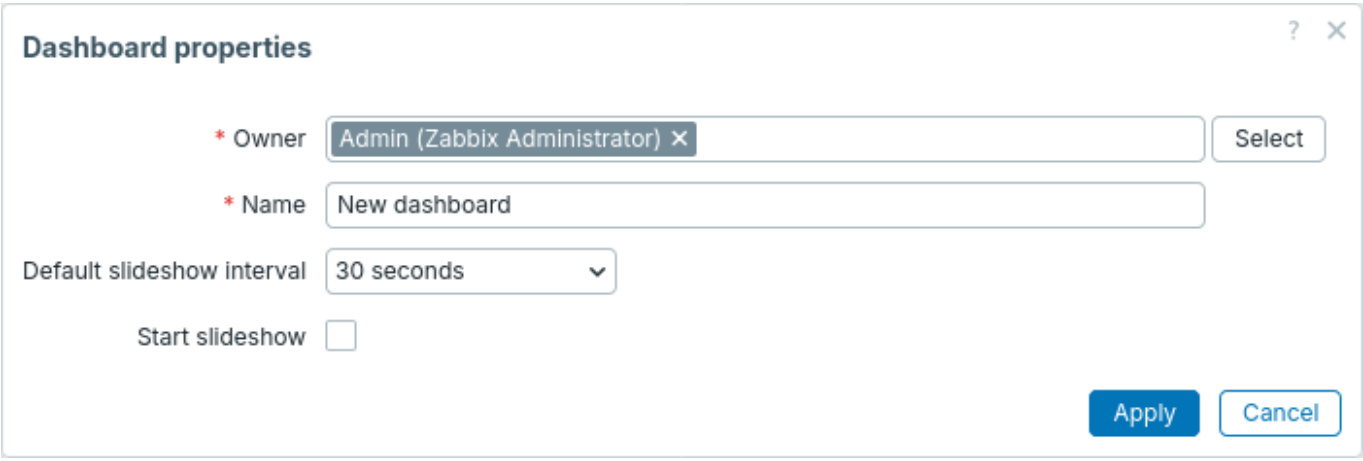
Zobacz Uprawnienia do dashboardów, aby dowiedzieć się, jak opcje udostępniania ograniczają działania dostępne dla użytkowników.

Tworzenie dashboardu

Dashboard można utworzyć na dwa sposoby:

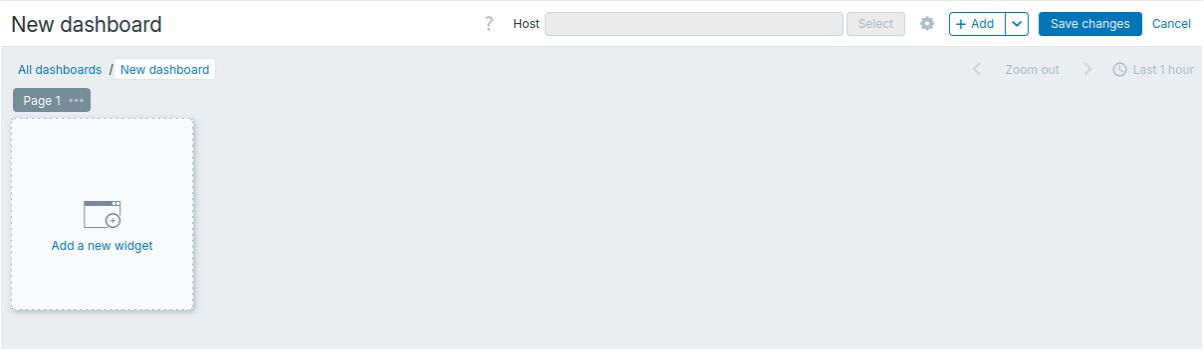
- Na liście dashboardów kliknij przycisk *Create dashboard*.
- W trybie wyświetlania dashboardu kliknij opcję *Create new* w menu *Actions* dashboardu.

Następnie określ właściwości dashboardu:



Parametr	Opis
Owner	Wybierz użytkownika, który będzie właścicielem dashboardu.
Name	Wprowadź nazwę dashboardu.
Default slideshow interval	Wybierz, jak długo każda strona dashboardu będzie wyświetlana, zanim pokaz slajdów przejdzie do następnej strony.
Start slideshow	Zaznacz to pole wyboru, aby pokaz slajdów uruchamiał się automatycznie, gdy dashboard będzie zawierał więcej niż jedną stronę.

Następnie kliknij przycisk *Apply*. Nowy dashboard zostanie otwarty w trybie edycji:



W tym miejscu możesz dodać widżety i strony dashboardu.

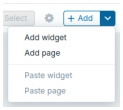
Pamiętaj, że minimalna szerokość dashboardu wynosi 1200 pikseli; w węższych oknach pojawi się poziomy pasek przewijania. Szerszy dashboard zawsze wypełnia okno przeglądarki; jego widżety nie mogą wykraczać poza krawędzie okna.

Kliknij przycisk *Save changes*, aby zapisać dashboard, lub *Cancel*, aby powrócić do poprzedniego ekranu.

Dodawanie widżetów

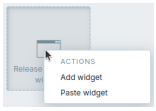
W trybie edycji dashboardu widżety można dodawać na dwa sposoby:

- Kliknij przycisk *Add*. Możesz również kliknąć strzałkę listy rozwijanej na przycisku *Add* i wybrać opcję *Add widget*.



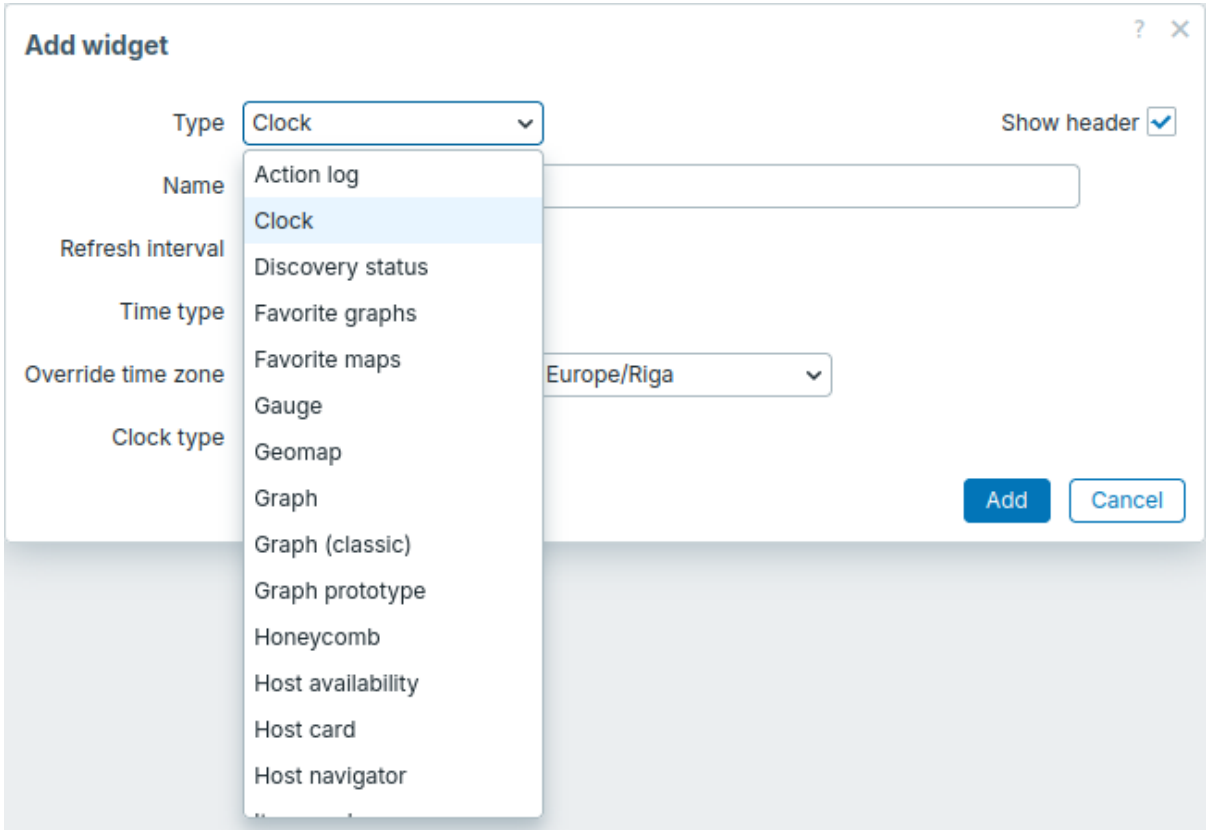
  Symbol zastępczy widżetu zostanie dodany w domyślnym rozmiarze do pierwszego dostępnego pustego miejsca na dashboardzie, a formularz konfiguracji widżetu zostanie otwarty.

- Umieść wskaźnik myszy nad pustym obszarem dashboardu. Następnie kliknij symbol zastępczy widżetu, aby otworzyć formularz konfiguracji widżetu. Możesz również kliknąć symbol zastępczy i przeciągnąć go, aby zmienić jego rozmiar, a następnie zwolnić przycisk myszy, aby otworzyć formularz konfiguracji widżetu; jeśli widżet został skopiowany do schowka, najpierw pojawi się monit o wybranie opcji *Add widget* lub *Paste widget*.



Następnie wykonaj następujące kroki:

1. W formularzu konfiguracji widżetu wybierz typ widżetu.
2. Skonfiguruj parametry widżetu.
3. Kliknij przycisk *Add*, aby dodać widżet do dashboardu.



Widżet zostanie dodany w domyślnym rozmiarze lub przeskalowany tak, aby dopasować go do dostępnego miejsca. Możesz zmieniać rozmiar i położenie widżetów zgodnie z opisem w sekcji *Edit widgets*.

Możesz również kopiować i wklejać widżety, aby tworzyć nowe widżety z taką samą konfiguracją, zarówno w obrębie tego samego dashboardu, jak i między dashboardami otwartymi w różnych kartach przeglądarki.

Kopiowanie/wklejanie widżetów

Aby skopiować i wkleić widżet:

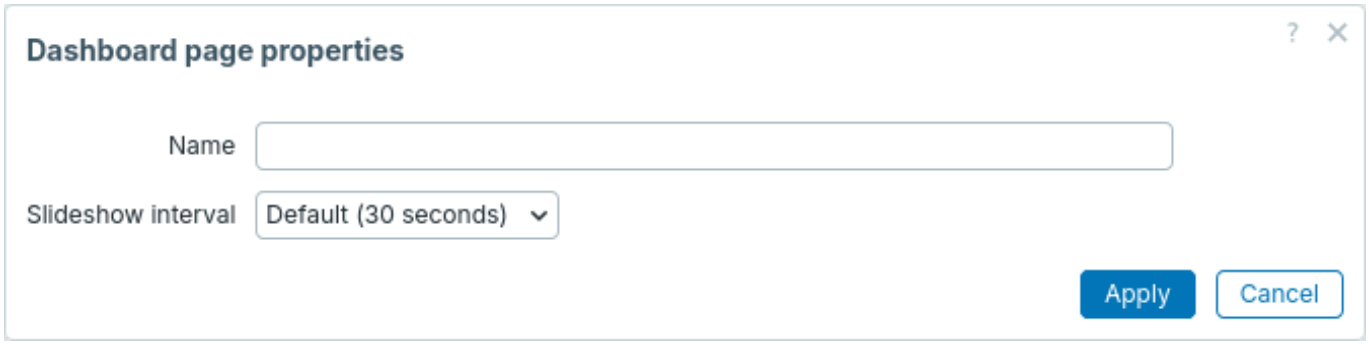
1. W menu *Actions* widżetu ⋮ wybierz opcję *Copy*.
2. W trybie edycji dashboardu kliknij strzałkę listy rozwijanej na przycisku *Add* i wybierz opcję *Paste widget*. Możesz również kliknąć pusty obszar dashboardu i wybrać opcję *Paste widget*.

Skopiowany widżet można także wkleić na istniejący widżet, korzystając z opcji *Paste* w jego menu *Actions*.

Dodawanie stron

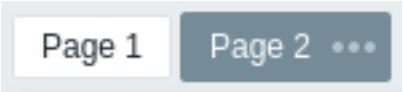
Aby dodać nową stronę do dashboardu:

1. W trybie edycji dashboardu kliknij strzałkę listy rozwijanej na przycisku *Add* i wybierz *Add page*.
2. Uzupełnij właściwości strony:
   - *Name* - wprowadź nazwę strony. Jeśli pozostawisz nazwę pustą, strona zostanie dodana z nazwą *Page N*, gdzie „N" oznacza kolejny numer strony.
   - *Slideshow interval* - wybierz, jak długo strona ma być wyświetlana, zanim pokaz slajdów przejdzie do następnej strony.



3. Kliknij *Apply*.

Dla nowej strony zostanie dodana nowa karta (np. *Page 2*).



Możesz również kopiować i wklejać strony, aby tworzyć nowe strony z tymi samymi właściwościami — zarówno w obrębie tego samego dashboardu, jak i między dashboardami otwartymi w różnych kartach.

Kopiowanie/wklejanie stron

Aby skopiować i wkleić stronę:

1. W menu *Działania* strony wybierz opcję *Kopiuj*.
2. W trybie edycji dashboardu kliknij strzałkę listy rozwijanej na przycisku *Dodaj* i wybierz opcję *Wklej stronę*.

Edycja dashboardu

Aby edytować dashboard, kliknij przycisk *Edit dashboard* w prawym górnym rogu. Możesz również kliknąć przycisk *Edit* dla widgetu. Dashboard zostanie otwarty w trybie edycji.

W tym miejscu możesz zarządzać dashboardami w następujący sposób:

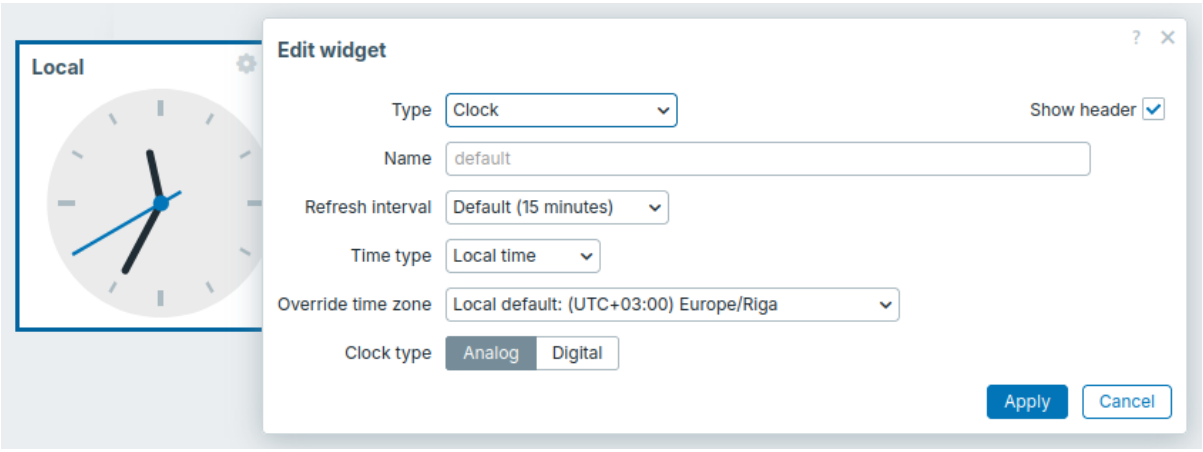
- Edytuj właściwości dashboardu, klikając przycisk koła zębatego.
- Edytuj widgety dashboardu.
- Uzyskaj dostęp do następujących działań dotyczących dashboardu, klikając strzałkę rozwijaną na przycisku *Add*:
  - *Add widget* - dodaj widget do dashboardu.
  - *Add page* - dodaj stronę do dashboardu.
  - *Paste widget* - wklej skopiowany widget. Ta opcja jest wyszarzona, jeśli nie skopiowano żadnego widgetu. Jednocześnie można skopiować tylko jeden widget.
  - *Paste page* - wklej skopiowaną stronę. Ta opcja jest wyszarzona, jeśli nie skopiowano żadnej strony. Jednocześnie można skopiować tylko jedną stronę.

Po zakończeniu edycji dashboardu kliknij przycisk *Save changes*, aby zapisać dashboard, lub *Cancel*, aby powrócić do poprzedniego ekranu.

Edytowanie widżetów

Widżet można edytować w trybie edycji dashboardu, klikając jego przycisk *Edit*. Alternatywnie, podczas wyświetlania dashboardu można umieścić kursor nad widżetem i kliknąć przycisk *Edit*.

Podczas edytowania widżet jest wyróżniony, a zmiany są wyświetlane w czasie rzeczywistym.



Widżet i jego formularz konfiguracji można przenosić, przeciągając ich paski tytułu, a także zmieniać rozmiar widżetu, przeciągając uchwyty zmiany rozmiaru.



Podczas edytowania widżetu w jego menu *Actions* dostępne są następujące działania:

- *Copy* - kopiowanie widżetu.
- *Paste* - wklejenie skopiowanego widżetu w miejsce tego widżetu. Ta opcja jest wyszarzona, jeśli nie skopiowano żadnego widżetu.
- *Delete* - usunięcie widżetu.

Jeśli wymagane parametry widżetu nie są skonfigurowane, widżet przechodzi do stanu *Widget is not fully configured*.

Po zakończeniu edycji kliknij przycisk *Apply* w formularzu konfiguracji widżetu, a następnie przycisk *Save changes* w prawym górnym rogu dashboardu, aby zastosować zmiany.

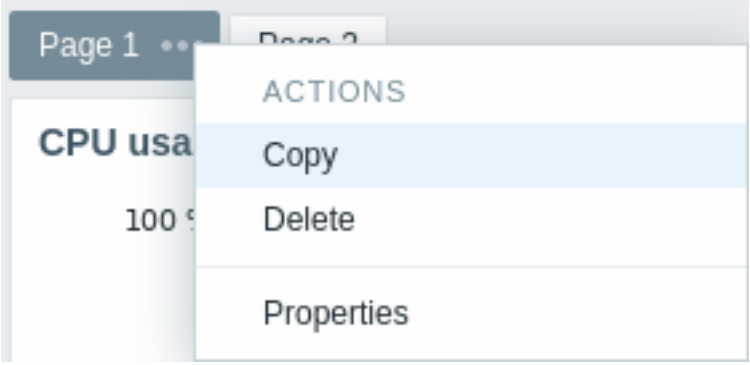
Edytowanie stron

Strony można edytować w trybie edycji dashboardu.

Kolejność stron można zmieniać, przeciągając i upuszczając karty stron. Zmiana kolejności nie powoduje zmiany nazw stron, z wyjątkiem stron z automatycznie przypisanymi nazwami (*Page N*), które są ponownie numerowane.

Podczas edytowania strony w menu *Actions* dostępne są następujące działania:

- *Copy* - kopiowanie strony.
- *Delete* - usunięcie strony.
- *Properties* - dostosowanie właściwości strony.



Uprawnienia do pulpitów nawigacyjnych

Uprawnienia do pulpitów nawigacyjnych dla zwykłych użytkowników i użytkowników typu *Admin* są ograniczone w sposób opisany poniżej, pod warunkiem że dostęp do pulpitów nawigacyjnych oraz opcja *Tworzenie i edytowanie pulpitów nawigacyjnych* są włączone dla przypisanej im roli użytkownika:

- Mogą wyświetlać i klonować publiczny pulpit nawigacyjny, nawet jeśli do ustawień udostępniania pulpitu nie dodano żadnych użytkowników ani grup użytkowników.
- Mogą wyświetlać i klonować prywatny pulpit nawigacyjny, jeśli mają do niego co najmniej uprawnienia *Odczyt*, zgodnie z ustawieniami udostępniania.
- Mogą edytować i usuwać pulpit nawigacyjny tylko wtedy, gdy mają do niego uprawnienia *Odczyt i zapis*, zgodnie z ustawieniami udostępniania.
- Nie mogą zmieniać właściciela pulpitu nawigacyjnego.

Widżety dashboardu

Przegląd

Zabbix udostępnia różnorodne widgety pulpitu, które wizualizują i podsumowują dane monitoringu, zapewniając przejrzysty przegląd wydajności i dostępności systemu.

Poniżej znajduje się przegląd wszystkich widgetów, pogrupowanych według ich głównego przeznaczenia.

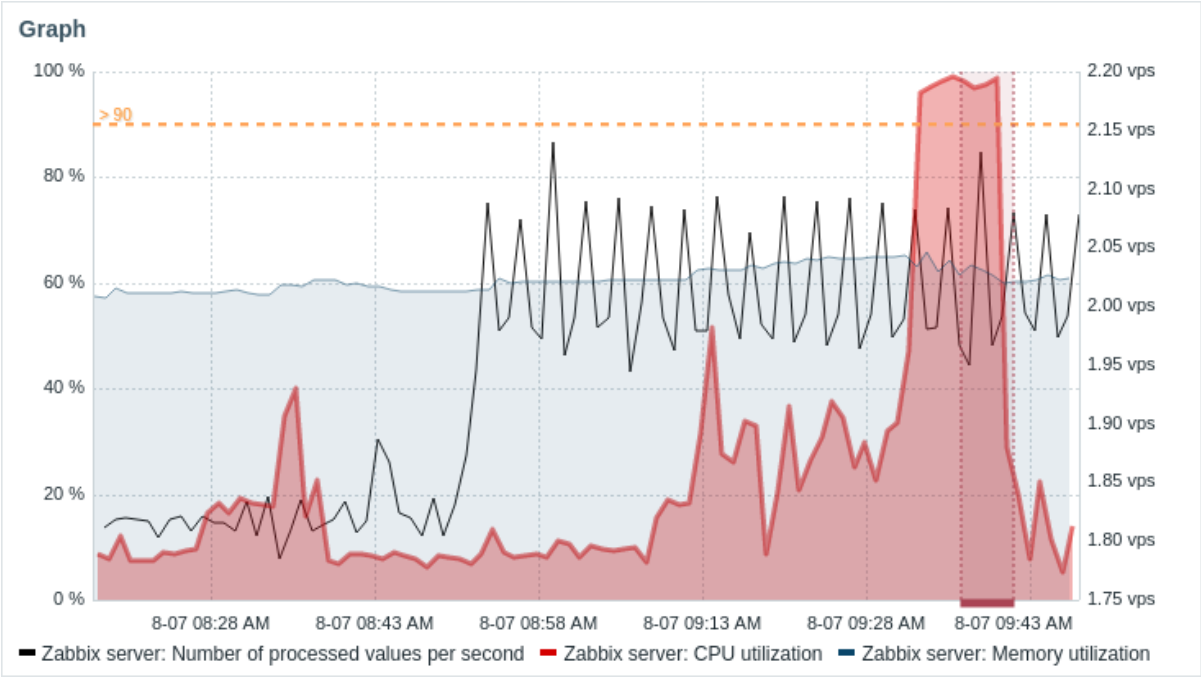
Szczegółowe informacje dotyczące konfiguracji widgetów można znaleźć w sekcji Parametry widgetów pulpitu.

Wizualizacja danych

Te widżety prezentują dane w różnych formatach wizualnych, pomagając monitorować dane i problemy na różne sposoby.

Wykres

Widżet Graph wyświetla dane numeryczne pozycji w postaci wykresu wektorowego. Może pomóc w śledzeniu metryk, wykrywaniu problemów oraz porównywaniu wartości w czasie i pomiędzy hostami.

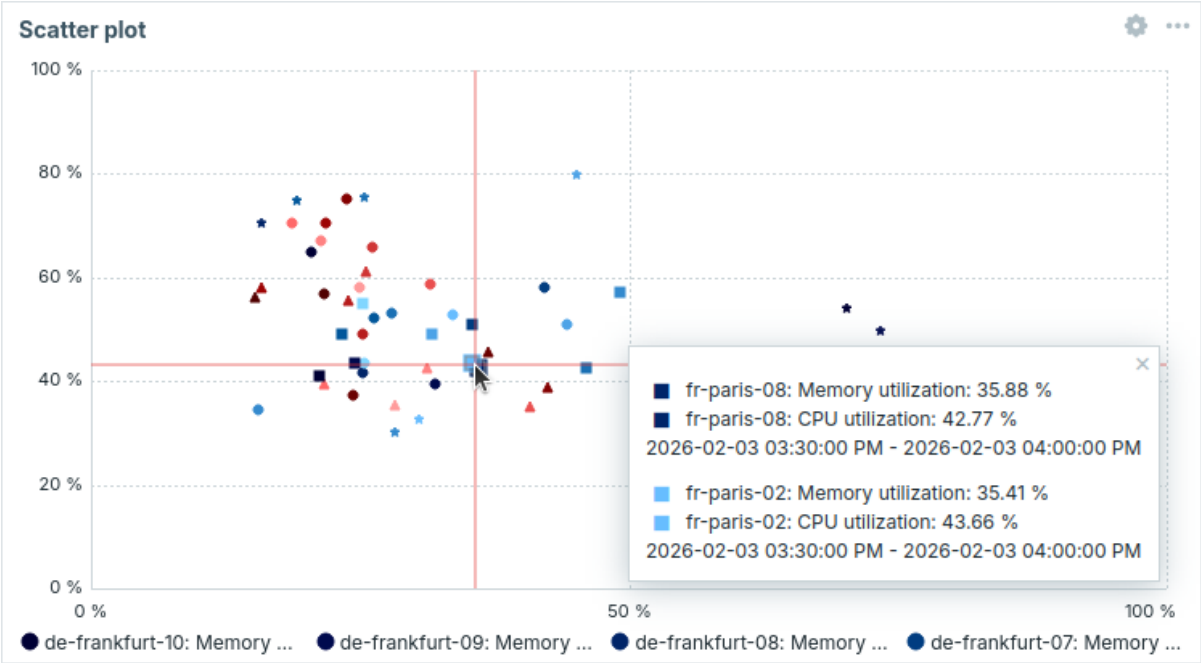


Zobacz także:

- Graph (classic) — starszy widżet Graph, który wyświetla dane numeryczne pozycji w postaci wykresu opartego na obrazie.
- Graph prototype — wyświetla siatkę automatycznie tworzonych niestandardowych wykresów na podstawie prototypów wykresów lub prototypów pozycji z reguły wykrywania niskiego poziomu.
- Favorite graphs — wyświetla alfabetyczną listę skrótów do wykresów oznaczonych przez bieżącego użytkownika jako ulubione.

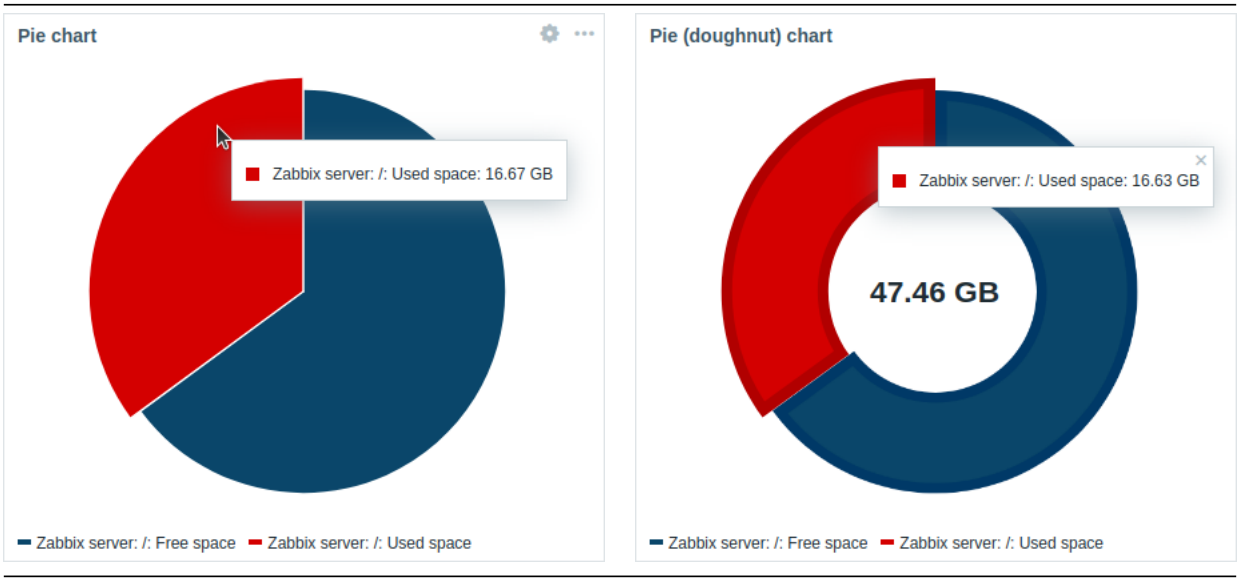
Wykres punktowy

Widżet Scatter plot wyświetla zależność między dwiema metrykami, nanosząc pojedyncze punkty danych wzdłuż osi X i Y. Pomaga to ujawnić wzorce, skupiska, korelacje i wartości odstające w zbiorze danych.



Wykres kołowy

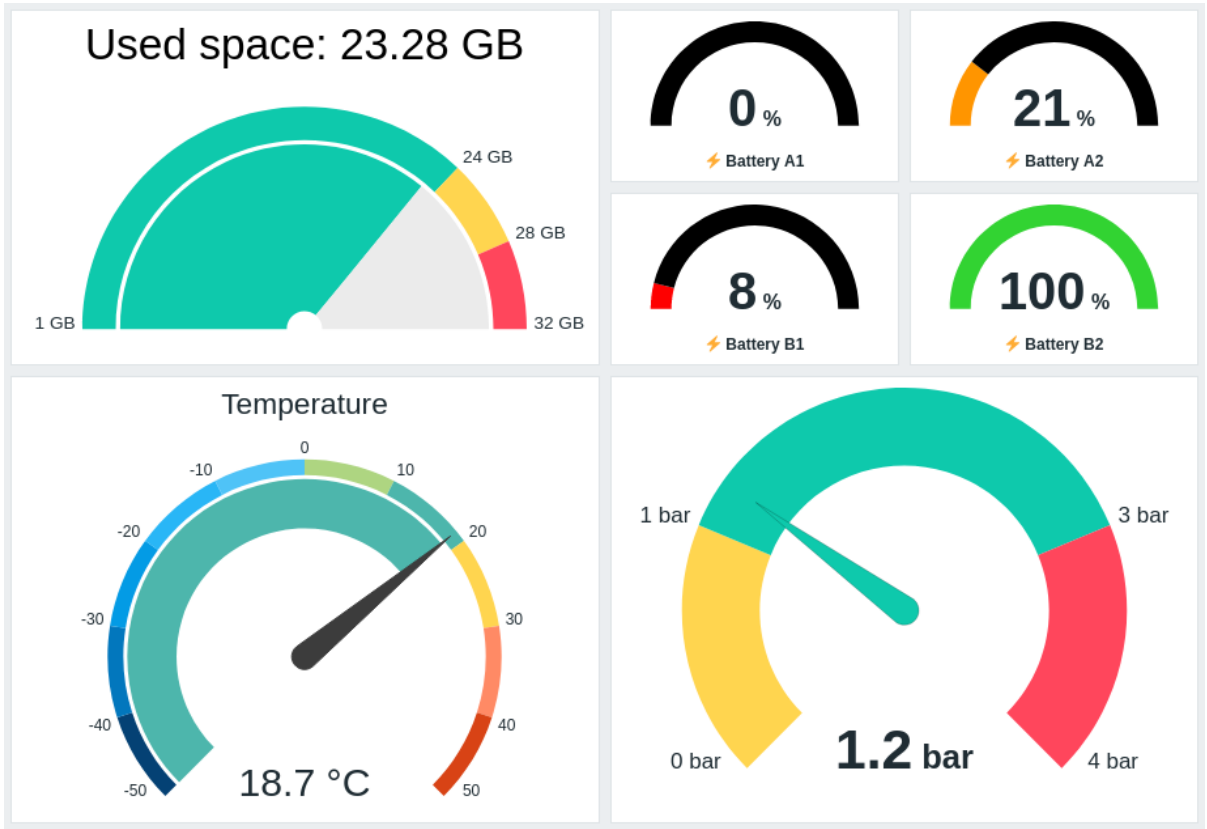
Widżet Wykres kołowy wyświetla numeryczne dane pozycji jako wektorowy wykres kołowy lub pierścieniowy. Ten widżet może pomóc zwizualizować, w jaki sposób pozycje lub hosty wpływają na cały zbiór danych.



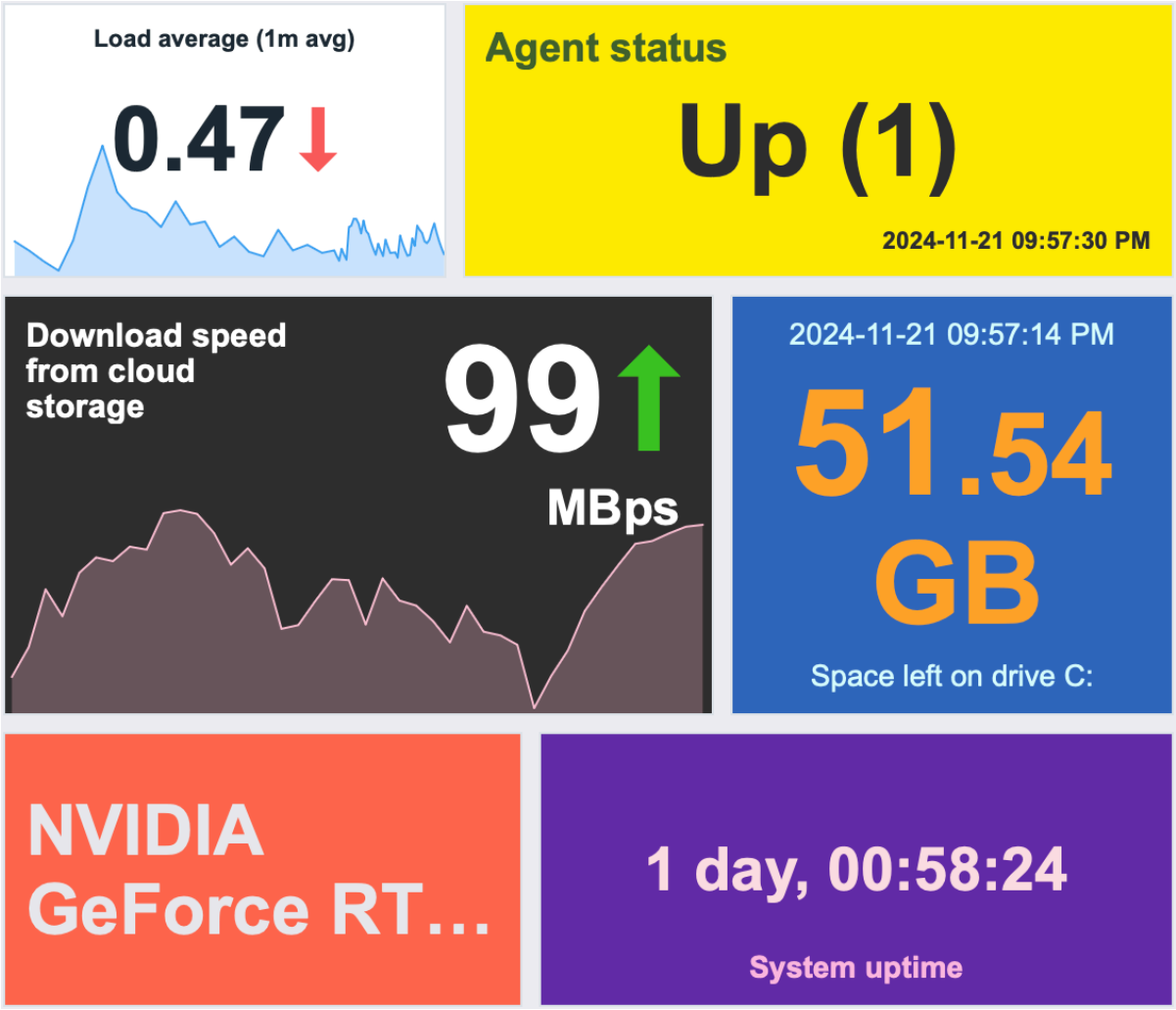
Wskaźnik/Wartość pozycji

Widżety Wskaźnik/Wartość pozycji wyświetlają wartość pojedynczej pozycji, co jest przydatne do śledzenia ważnych metryk, wizualizacji progów oraz wykrywania nagłych zmian w danych.

Widżet Wskaźnik:

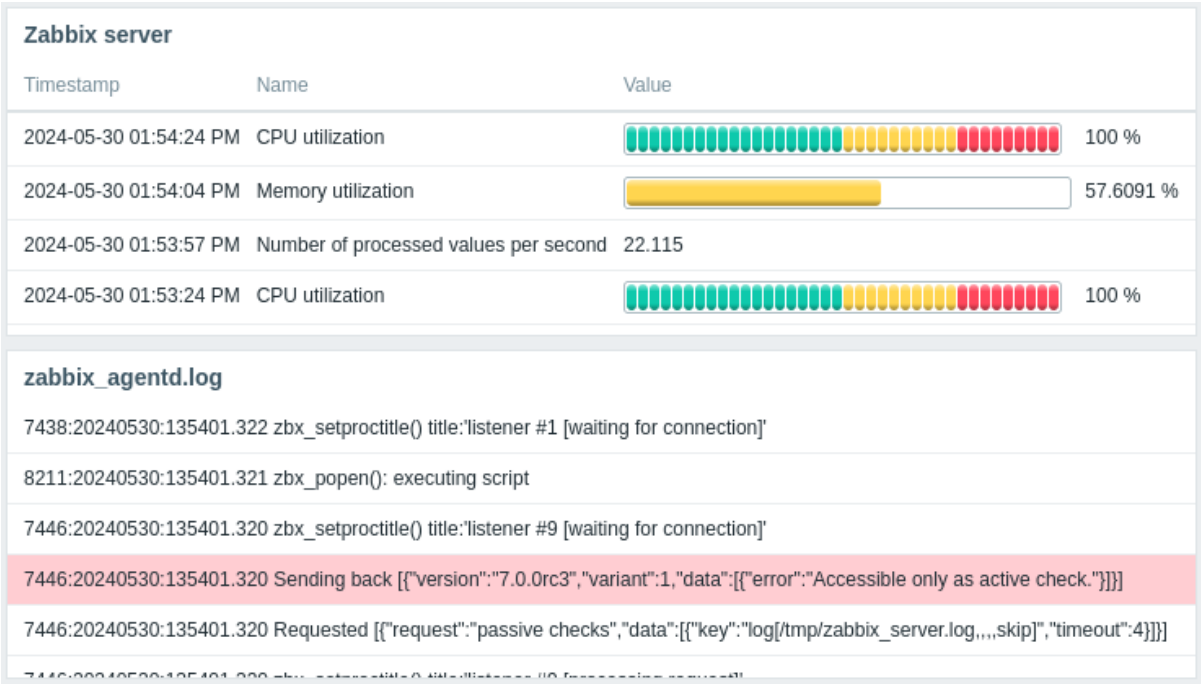


Widżet Wartość pozycji:



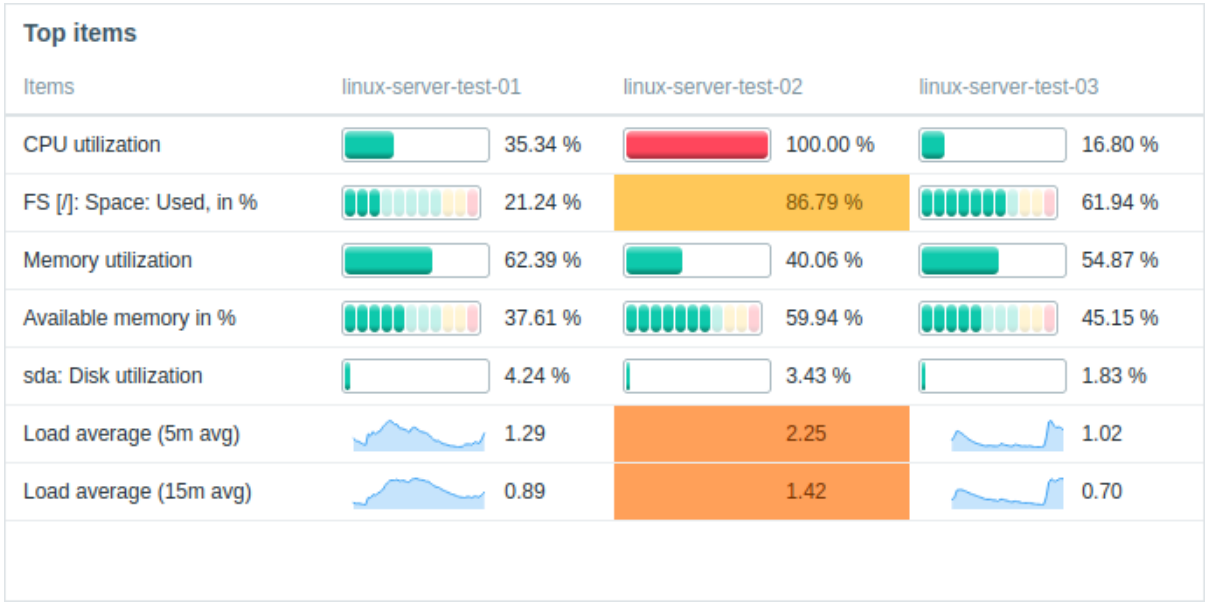
Historia pozycji

Widżet Historia pozycji wyświetla najnowsze dane dla wszystkich typów pozycji (numerycznych, tekstowych itp.) w formacie tabeli. Może wyświetlać paski postępu i obrazy (przydatne w przypadku pozycji przeglądarki) oraz wyróżniać wartości (przydatne przy monitorowaniu plików dziennika).



Najlepsze pozycje

Widżet Top items wyświetla najwyższe/najniższe wartości dla wybranych pozycji, zapewniając szybki przegląd ich wydajności.

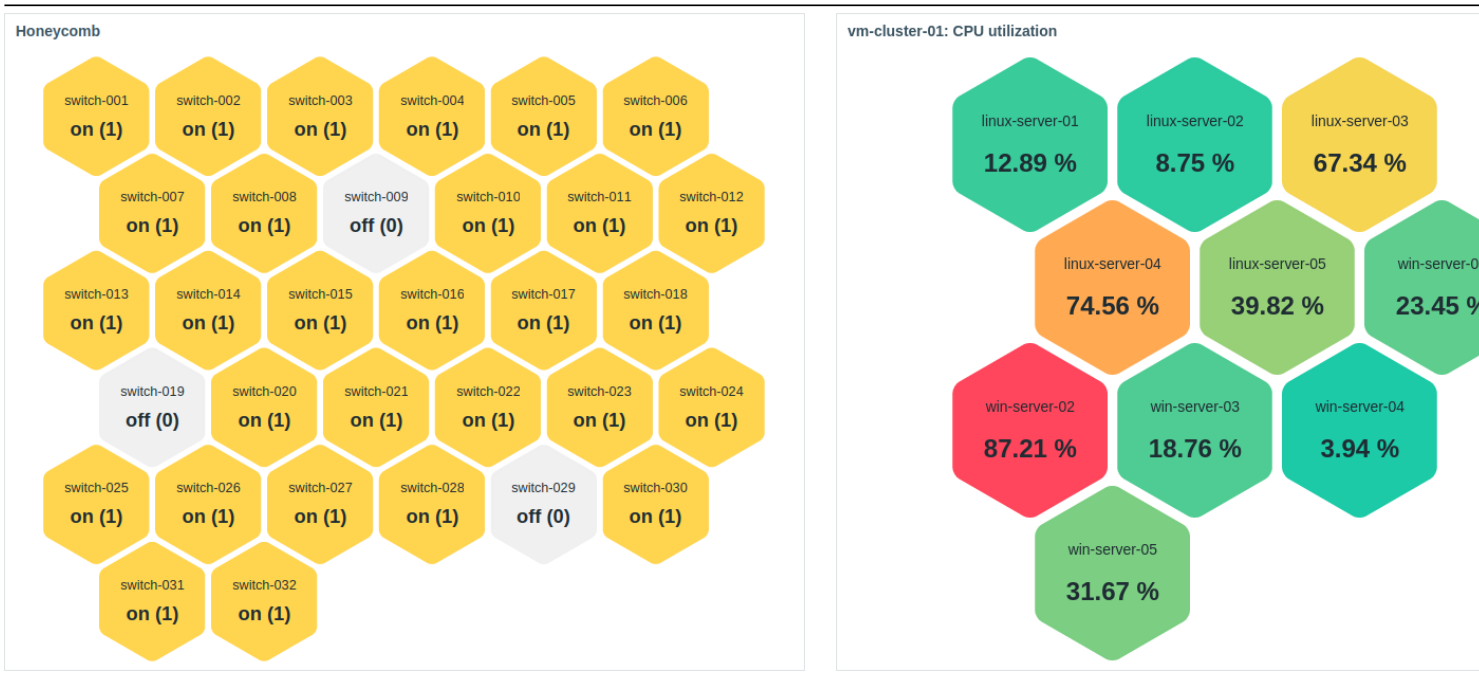


Zobacz także:

- Top hosts — wyświetla najwyższe/najniższe wartości dla wybranych pozycji z wielu hostów.
- Top triggers — wyświetla wyzwalacze z największą liczbą problemów.

Honeycomb

Widżet Honeycomb wyświetla hosty lub pozycje jako sześciokątne komórki w siatce. Ten układ ułatwia wykrywanie hostów/pozycji z problemami, dostrzeganie skupisk problemów, porównywanie grup na pierwszy rzut oka i nie tylko.

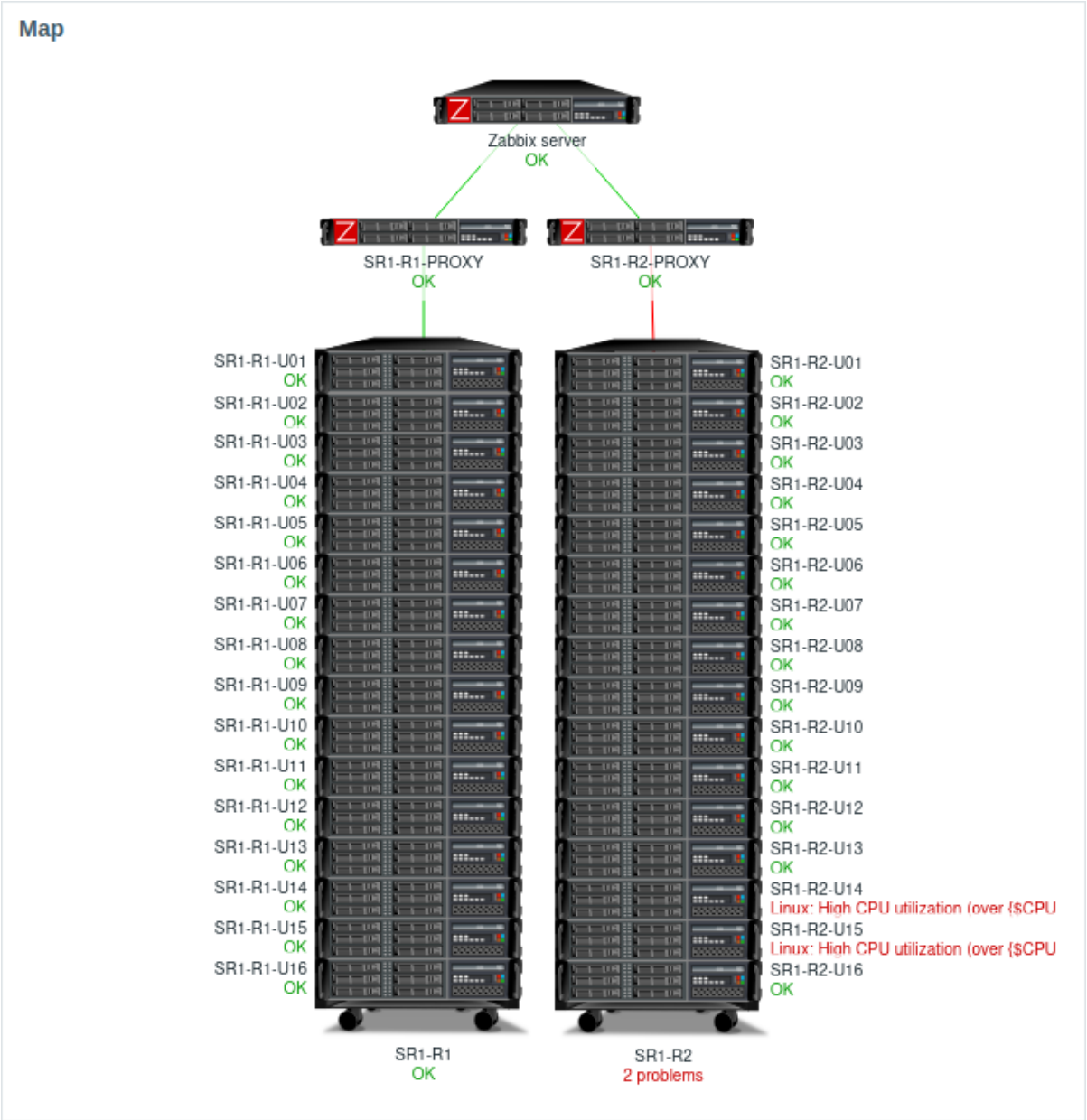


Mapy

Widżety map wizualizują topologię sieci lub lokalizacje hostów, pomagając ocenić ogólny stan oraz stan w poszczególnych lokalizacjach.

Mapa

Widżet Mapa może wyświetlać mapę sieci lub podobną wizualizację, zapewniając dynamiczny przegląd sieci.

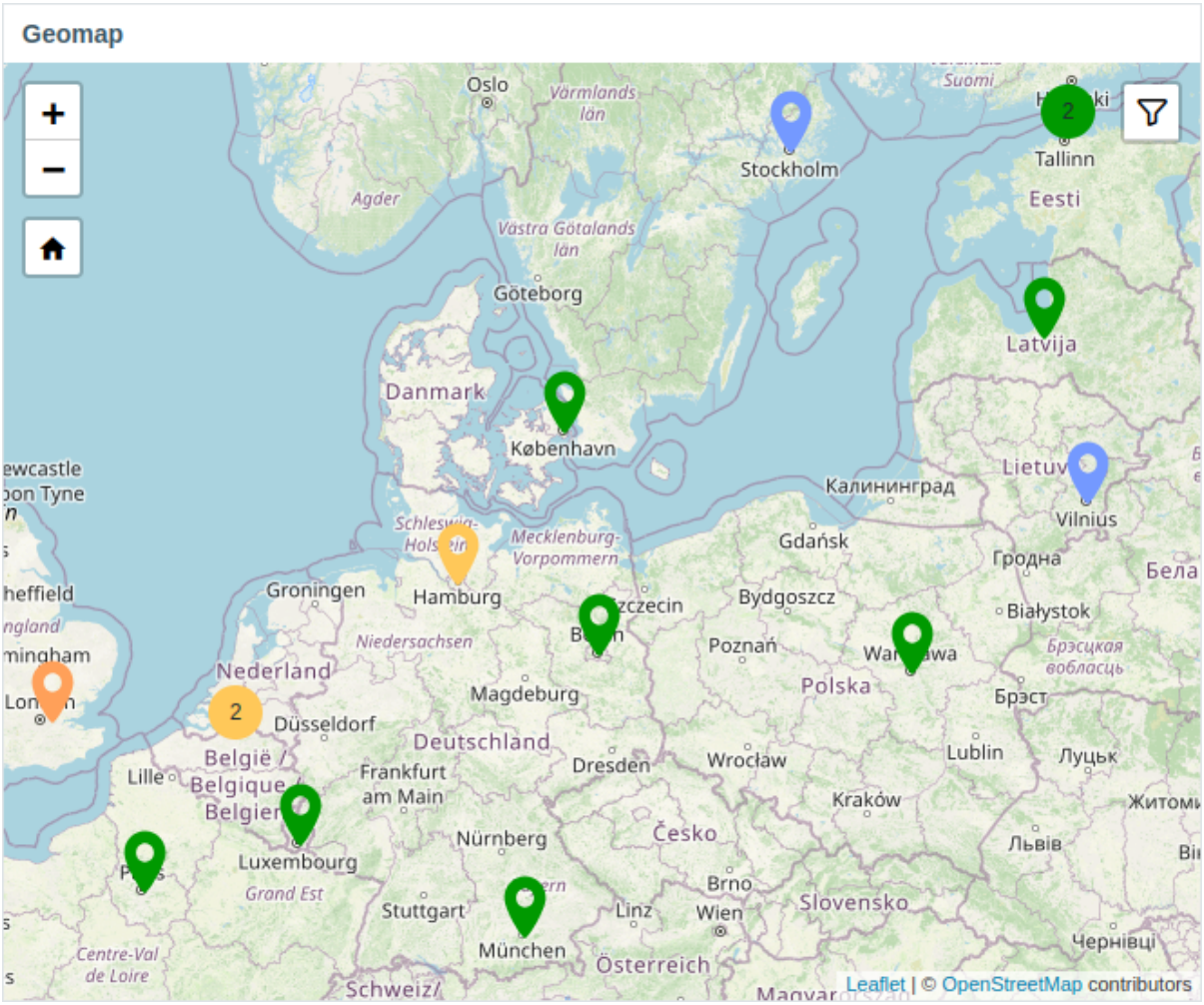


Zobacz także:

- Drzewo nawigacji map — wyświetla hierarchię istniejących map oraz liczbę problemów dla każdej mapy i grupy map.
- Ulubione mapy — wyświetla listę skrótów do map oznaczonych przez bieżącego użytkownika jako ulubione, posortowaną alfabetycznie.

Geomap

Widżet Geomap wyświetla hosty jako znaczniki na interaktywnej mapie opartej na Leaflet, przy czym kolory znaczników wskazują najpoważniejszy problem każdego hosta.

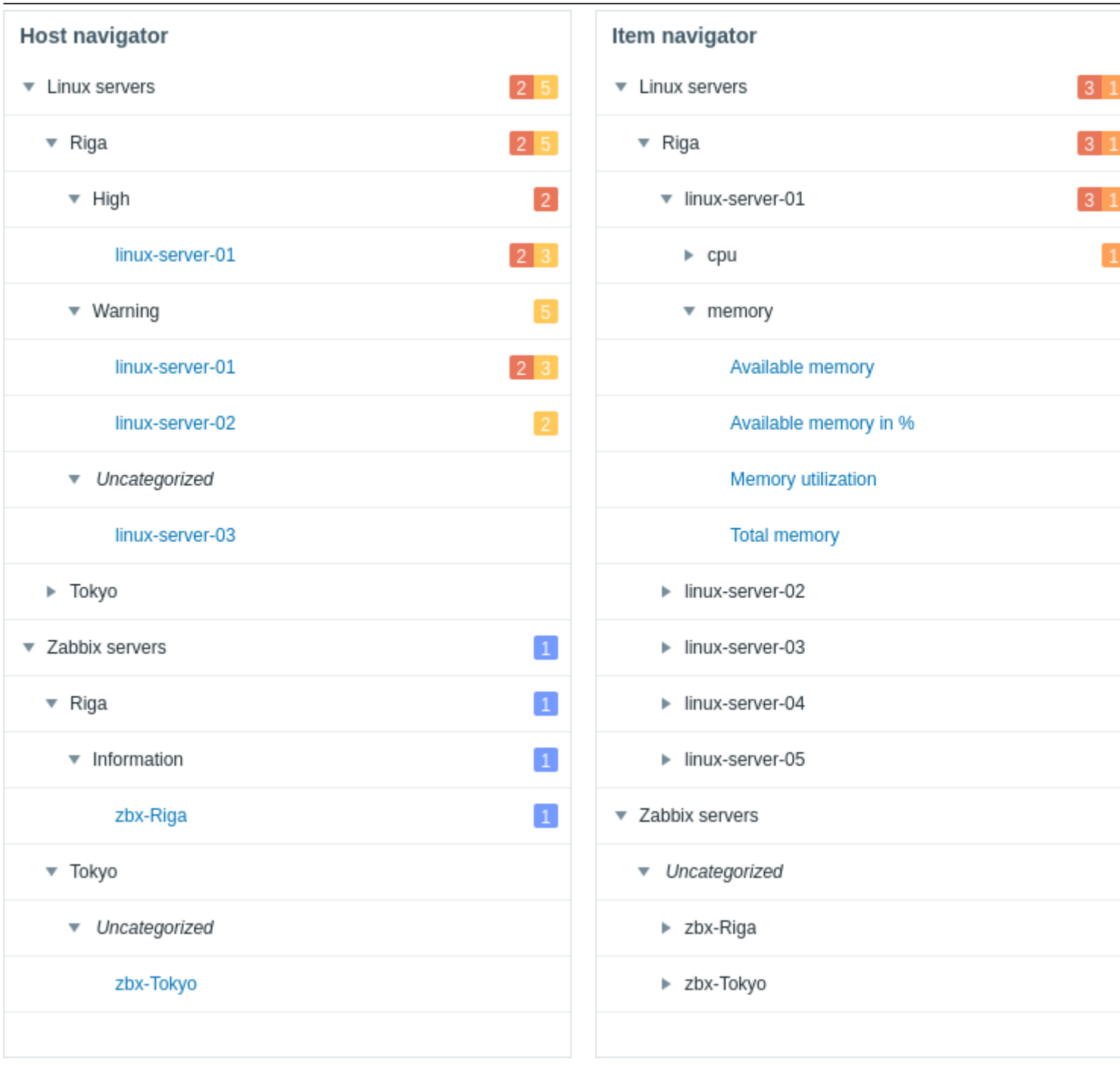


Szczegóły hosta/pozycji

Te widgety udostępniają listy oraz szczegółowe widoki hostów lub pozycji.

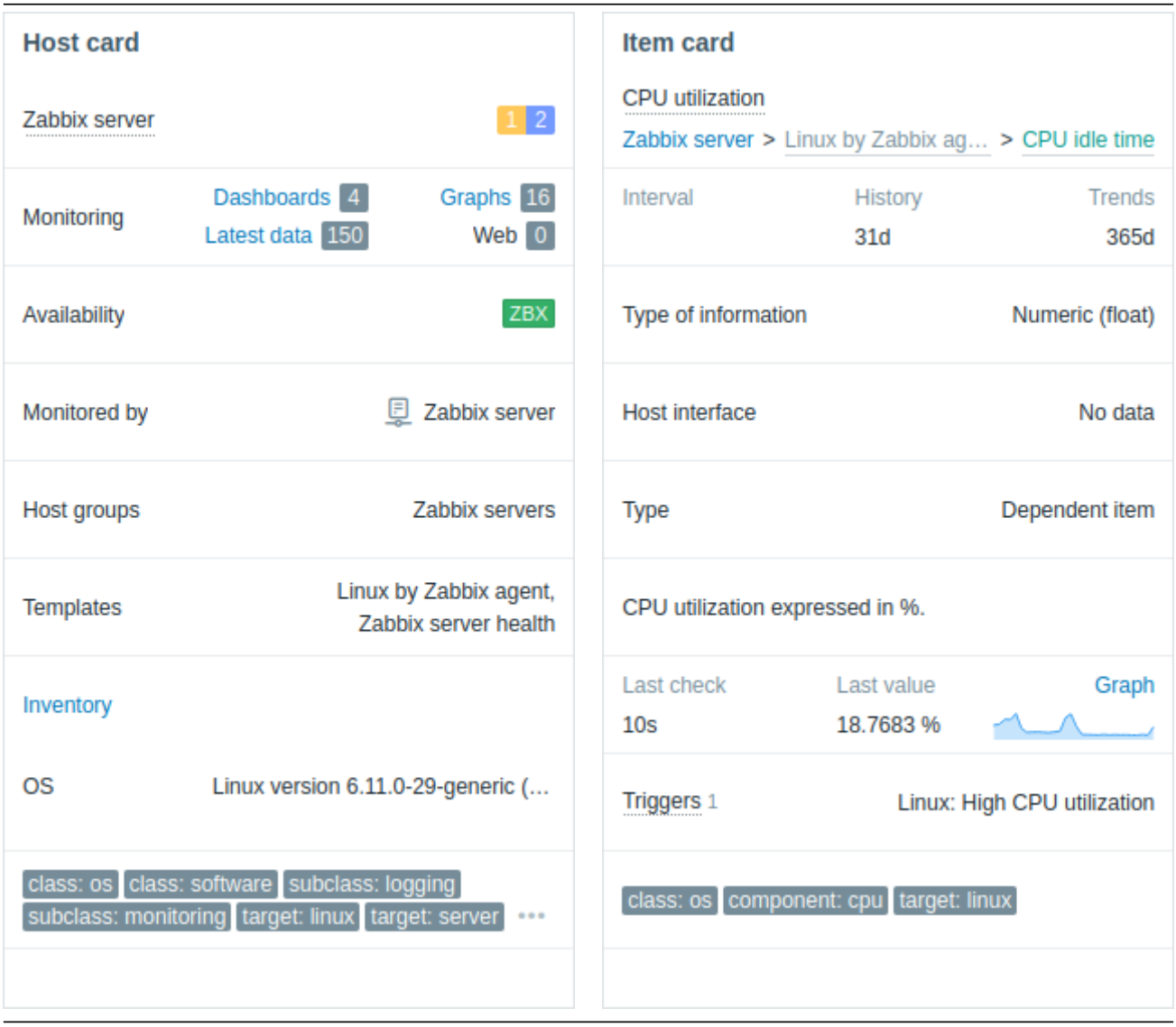
Nawigator hostów/pozycji

Widżety Nawigator hostów/Nawigator pozycji wyświetlają listę hostów/pozycji na podstawie różnych opcji filtrowania i grupowania. Te widżety są najbardziej przydatne do sterowania tym, co wyświetlają inne widżety, na podstawie wybranego hosta/pozycji.



Karta hosta/pozycji

Widżety Karta hosta/Karta pozycji wyświetlają szczegóły pojedynczego hosta/pozycji, umożliwiając szybkie sprawdzenie stanu i konfiguracji hosta lub pozycji (dostępność hosta, problemy, inwentarz, najnowsze dane pozycji, błędy, opis itp.).

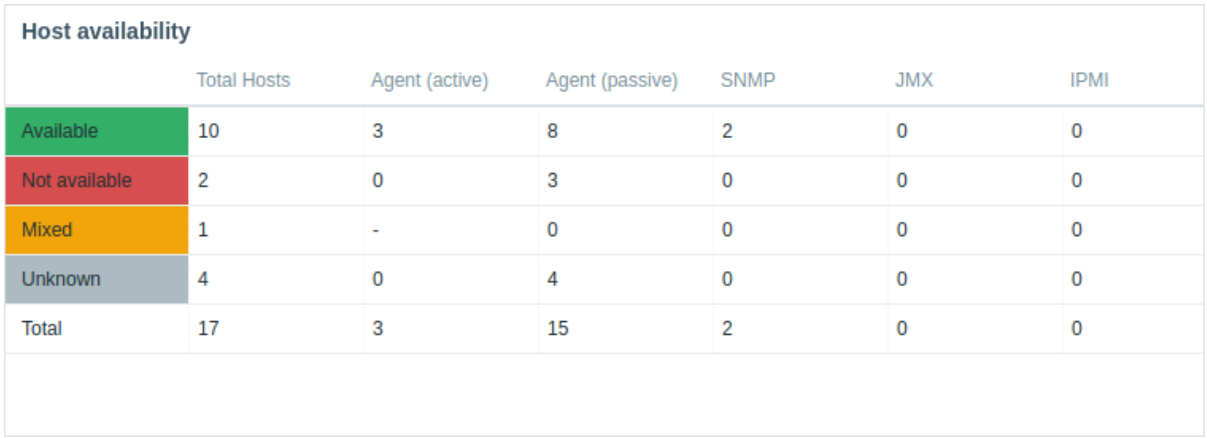


Status i raporty

Te widgety wyświetlają dane o stanie (np. dostępność, wyzwalacze, problemy), a także raporty.

Dostępność hostów

Widżet Dostępność hostów wyświetla dostępność hostów w wybranych grupach hostów, umożliwiając monitorowanie liczby hostów, które są dostępne lub niedostępne.

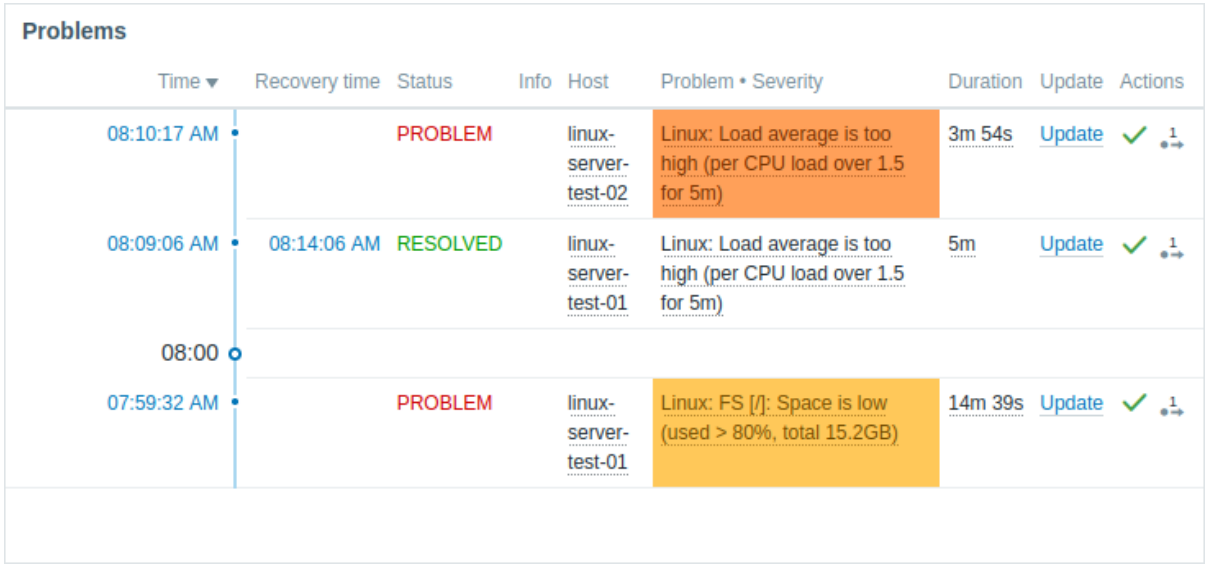


Host availability

	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

Problemy

Widżet Problems wyświetla bieżące problemy filtrowane według różnych parametrów (grupy hostów, hosty, nazwy problemów itp.), zapewniając przejrzysty widok tego, co wymaga uwagi.



Zobacz także:

- Problems by severity — wyświetla liczbę problemów przyczyn pogrupowanych według ważności.
- Problem hosts — wyświetla liczbę problemów przyczyn na grupę hostów i pokazuje najwyższą ważność problemu w każdej grupie.

Przegląd wyzwalaczy

Widżet Trigger overview wyświetla bieżące stany wyzwalaczy hostów w postaci tabeli z kolorowymi blokami. Zapewnia szybki wizualny podgląd kondycji i aktywności wyzwalaczy na różnych hostach.



Dziennik działań

Widżet Action log wyświetla szczegóły operacji (powiadomień, zdalnych poleceń) wykonywanych w ramach działania. Szczegółowy dziennik pomaga w audycie, rozwiązywaniu problemów i monitorowaniu wykonania każdego działania.

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

Raport SLA

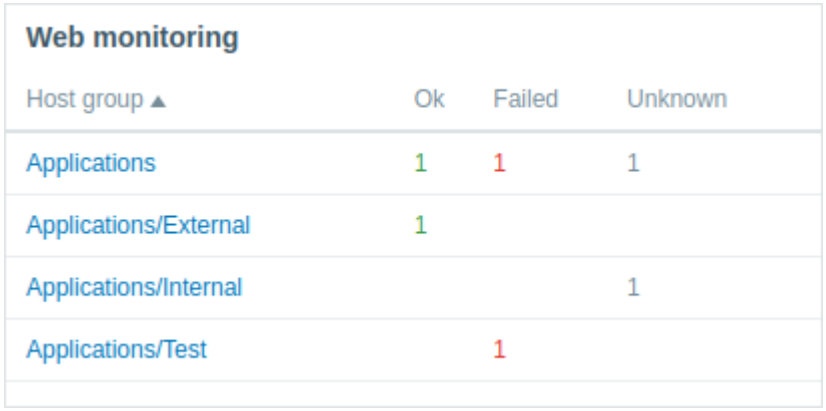
Widget Raport SLA wyświetla raporty SLA, pomagając monitorować wydajność usług względem zdefiniowanych celów.

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

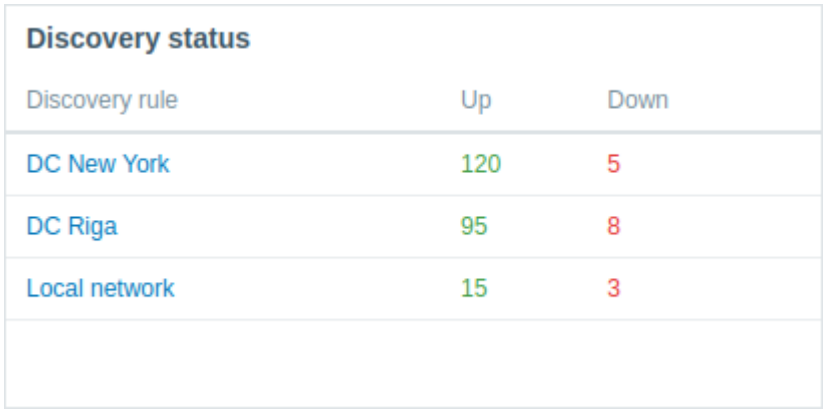
Monitorowanie WWW

Widżet Monitorowanie WWW wyświetla status aktywnych scenariuszy WWW.



Status wykrywania

Widget Discovery status wyświetla status urządzeń wykrytych przez włączone reguły wykrywania sieci. Pomaga monitorować, czy wykrywanie sieci działa prawidłowo, oraz liczbę urządzeń, które są dostępne lub niedostępne.



Narzędzia pulpitu i informacje o systemie

Te widżety dostarczają dodatkowych informacji i narzędzi, które wzbogacają pulpity.

Informacje o systemie

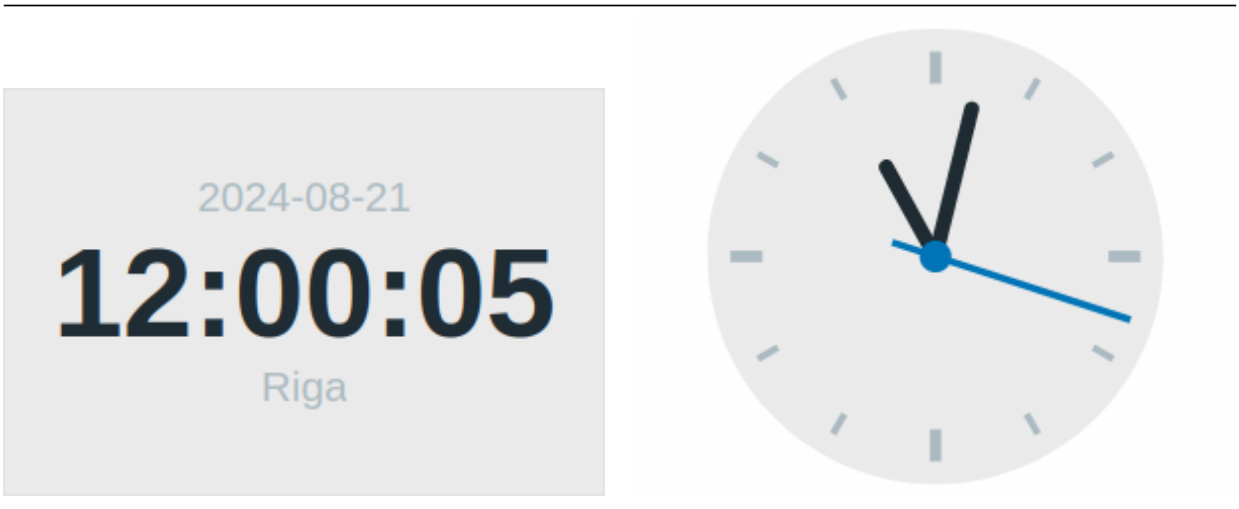
Widget Informacje o systemie wyświetla podsumowanie kluczowych danych serwera Zabbix i systemu lub szczegóły węzłów wysokiej dostępności.

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	

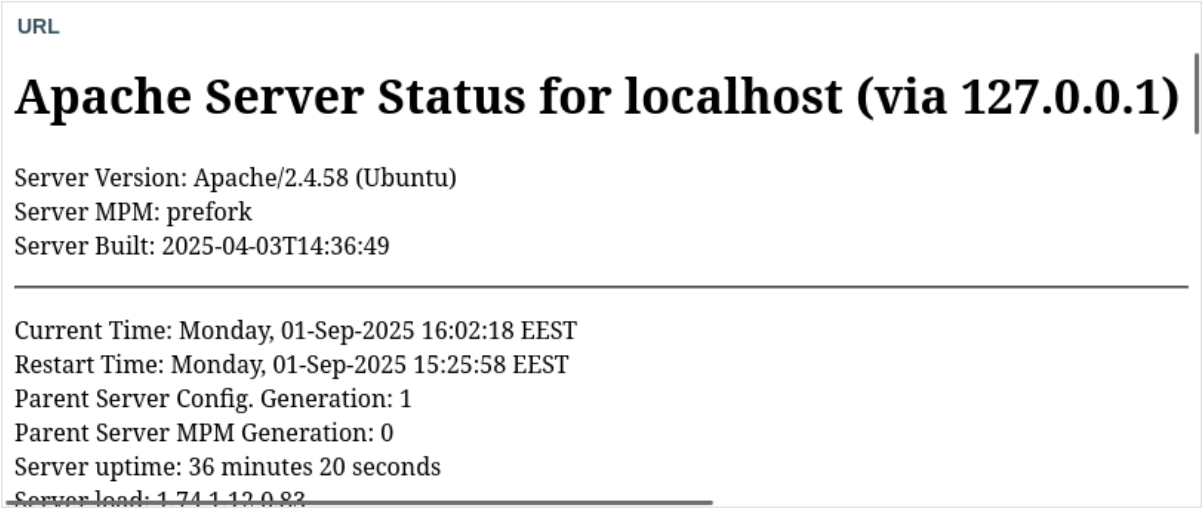
Zegar

Widget Clock wyświetla czas lokalny, czas serwera lub hosta w formacie analogowym lub cyfrowym.



URL

Widżet URL wyświetla zawartość pobraną z adresu URL.



Parametry widżetu dashboardu

Przegląd

Ta strona opisuje parametry wspólne dla wszystkich widgetów pulpitu, a także parametry dynamiczne, które umożliwiają wielu widgetom wymianę danych konfiguracyjnych między sobą lub z pulpitem.

Aby zobaczyć parametry specyficzne dla każdego widgetu, skorzystaj z panelu bocznego, aby przejść do odpowiedniej strony widgetu.

Wspólne parametry

Poniższe parametry są wspólne dla każdej pozycjii:

Nazwa	Wprowadź nazwę pozycji.
Interwał odświeżania	Skonfiguruj domyślny interwał odświeżania. Domyślne interwały odświeżania dla pozycji wynoszą od *Brak odświeżania* do *15 minut*, w zależności od typu pozycji. Na przykład: - *Brak odświeżania* dla pozycji *URL*; - *1 minuta* dla pozycji *Dziennik działań*; - *15 minut* dla pozycji *Zegar*. Interwały odświeżania można ustawić jako wartość domyślną dla wszystkich użytkowników. Przełącz pulpit na tryb edycji, kliknij przycisk edytowania pozycji i wybierz żądany interwał odświeżania z listy rozwijanej. Każdy użytkownik może również ustawić własny interwał odświeżania pozycji. W trybie wyświetlania pulpitu kliknij przycisk z trzema kropkami na pozycji i wybierz żądany interwał odświeżania z listy rozwijanej. Należy pamiętać, że unikatowy interwał odświeżania użytkownika ma pierwszeństwo przed ustawieniem pozycji i zostaje zachowany nawet po zmodyfikowaniu ustawienia pozycji.
Pokaż nagłówek	Zaznacz pole wyboru, aby stale wyświetlać nagłówek pozycji. Po usunięciu zaznaczenia nagłówek jest ukrywany w celu zaoszczędzenia miejsca i staje się widoczny tylko po najechaniu kursorem myszy na pozycję (zarówno w trybie wyświetlania, jak i edycji). Nagłówek jest również częściowo widoczny podczas przeciągania pozycji w nowe miejsce.

Parametry dynamiczne

Wiele parametrów widżetów (np. *Hosty*, *Nadpisz hosta*, *Okres czasu*) obsługuje dynamiczne źródła danych. Zamiast konfigurować wartości statyczne, możesz powiązać te parametry z innymi widżetami lub samym pulpitem. Widżety są wtedy automatycznie aktualizowane na podstawie wyborów lub rozgłoszeń z innych części pulpitu.

Zgodność widgetów

Niektóre widgety mogą rozgłaszać dane konfiguracyjne do innych widgetów, niektóre mogą nasłuchiwać danych, a niektóre mogą robić jedno i drugie. Na przykład:

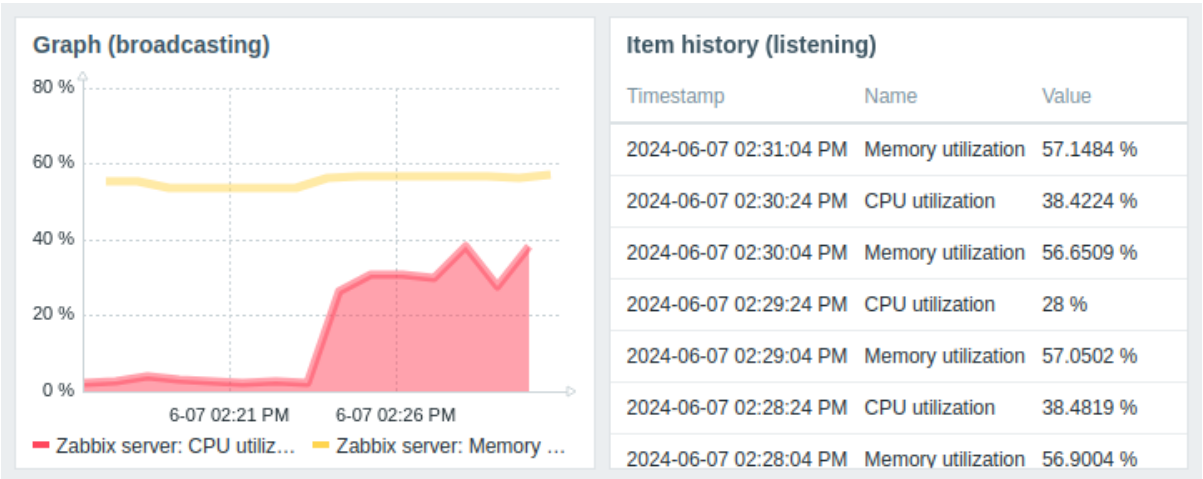
- Widget *Action log* może pobierać dane okresu tylko z widgetów *Graph*, *Graph (classic)* i *Graph prototype*.
- Widget *Geomap* może rozgłaszać dane hostów do widgetów, które ich nasłuchują (*Honeycomb*, *Top items* itd.), a także nasłuchiwać danych grup hostów i hostów z widgetów, które je rozgłaszają (*Honeycomb*, *Problem hosts* itd.).
- Widget *Clock* nie może rozgłaszać ani nasłuchiwać żadnych danych.

Poniższa tabela przedstawia możliwości rozgłaszania i nasłuchiwania dla każdego widgetu.

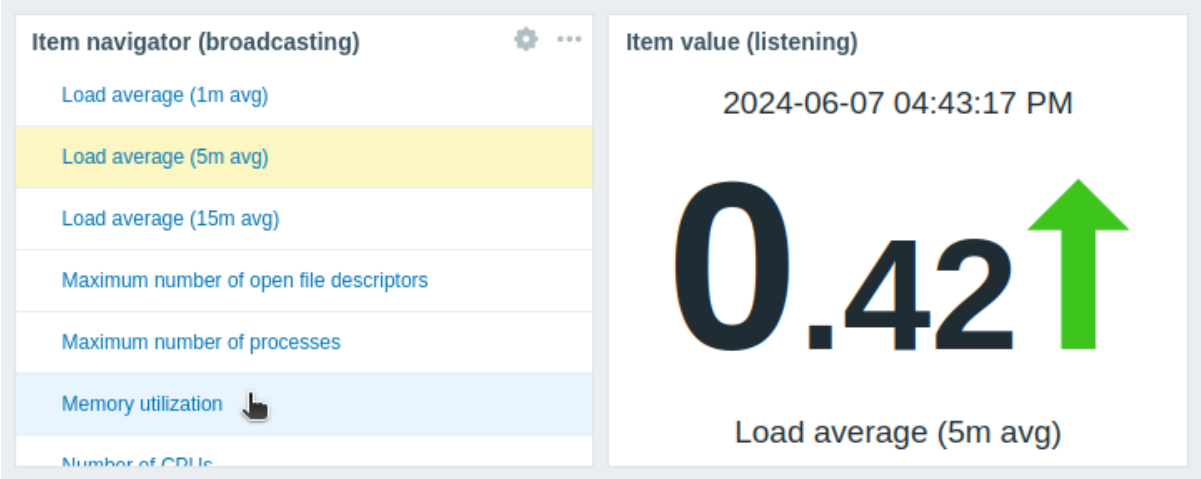
Widget	Rozgłasza	Nasłuchuje
Action log	-	Okres
Clock	-	-
Discovery status	-	-
Favorite graphs	-	-
Favorite maps	-	-
Gauge	-	Hosty, Pozycje
Geomap	Hosty	Grupy hostów, Hosty
Graph	Okres, Pozycje	Hosty, Pozycje, Okres
Graph (classic)	Okres	Hosty, Pozycje, Wykresy, Okres
Graph prototype	Okres	Hosty, Okres
Honeycomb	Hosty, Pozycje	Grupy hostów, Hosty
Host availability	-	Grupy hostów
Host card	-	Hosty
Host navigator	Hosty	Grupy hostów
Item card	-	Hosty, Pozycje, Okres
Item history	Pozycje	Hosty, Okres
Item navigator	Pozycje	Grupy hostów, Hosty
Item value	-	Hosty, Pozycje, Okres
Map	Grupy hostów, Hosty	Mapy
Map navigation tree	Mapy	-
Pie chart	-	Pozycje, Okres
Problem hosts	Grupy hostów	Grupy hostów, Hosty
Problems	Zdarzenia	Grupy hostów, Hosty
Problems by severity	Grupy hostów	Grupy hostów, Hosty
Scatter plot	Okres, Pozycje	Hosty, Pozycje, Okres
SLA report	-	-
System information	-	-
Top hosts	Hosty	Grupy hostów, Hosty
Top items	-	Grupy hostów, Hosty
Top triggers	-	Okres
Trigger overview	-	Grupy hostów, Hosty
URL	-	Hosty
Web monitoring	Grupy hostów	Grupy hostów, Hosty

Widżety różnią się sposobem, w jaki **rozgłaszają** dane do innych widżetów.

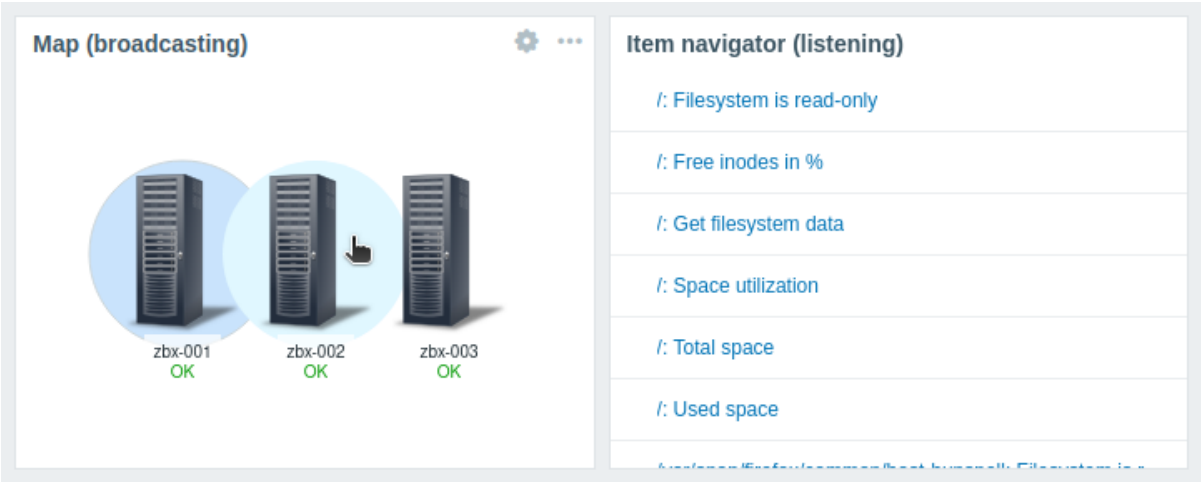
Widżety obsługujące rozgłaszanie zaczynają robić to automatycznie po utworzeniu. Na przykład widżet *Wykres* natychmiast rozgłasza dane okresu czasu do nasłuchujących widżetów.



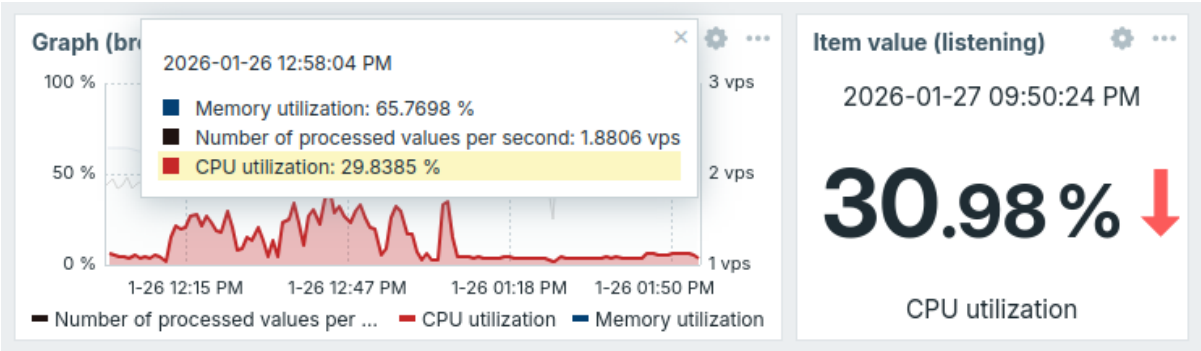
Widżety obsługujące wybór encji automatycznie rozgłaszają dane dla pierwszej dostępnej encji. Na przykład widżet *Nawigator pozycji* rozgłasza dane dla pierwszej pozycji na swojej liście pozycji. Po wybraniu innej pozycji rozgłasza dane dla tej pozycji. Po najechaniu kursorem pozycja jest podświetlana na jasnoniebiesko; po wybraniu jest podświetlana na żółto.



Widżet *Mapa* zachowuje się podobnie, rozgłaszając dane dla pierwszego elementu znajdującego się najbliżej lewego górnego rogu widocznego obszaru widżetu (w przypadku widżetu *Geomapa* — najbliżej środka). Po wybraniu innego elementu rozgłasza dane dla tego elementu. Po najechaniu kursorem element jest podświetlany na jasnoniebiesko; po wybraniu jest podświetlany na ciemnoniebiesko.

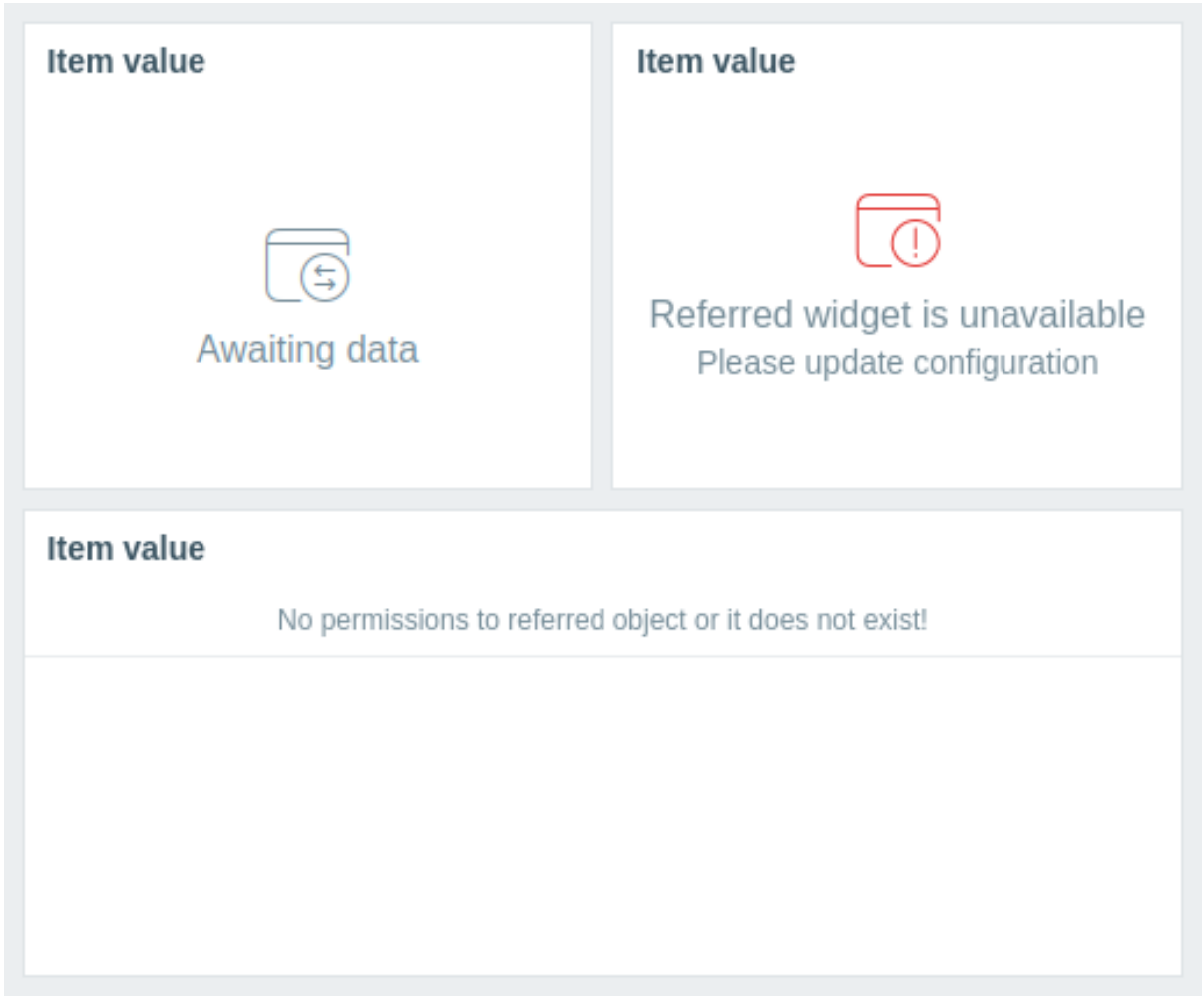


Widżety takie jak *Wykres* i *Wykres rozrzutu* mogą rozgłaszać dane pozycji po wybraniu pozycji w ich podpowiedziach:



Widżety mają również określone zachowania podczas **nasłuchiwania** danych z innych widżetów:

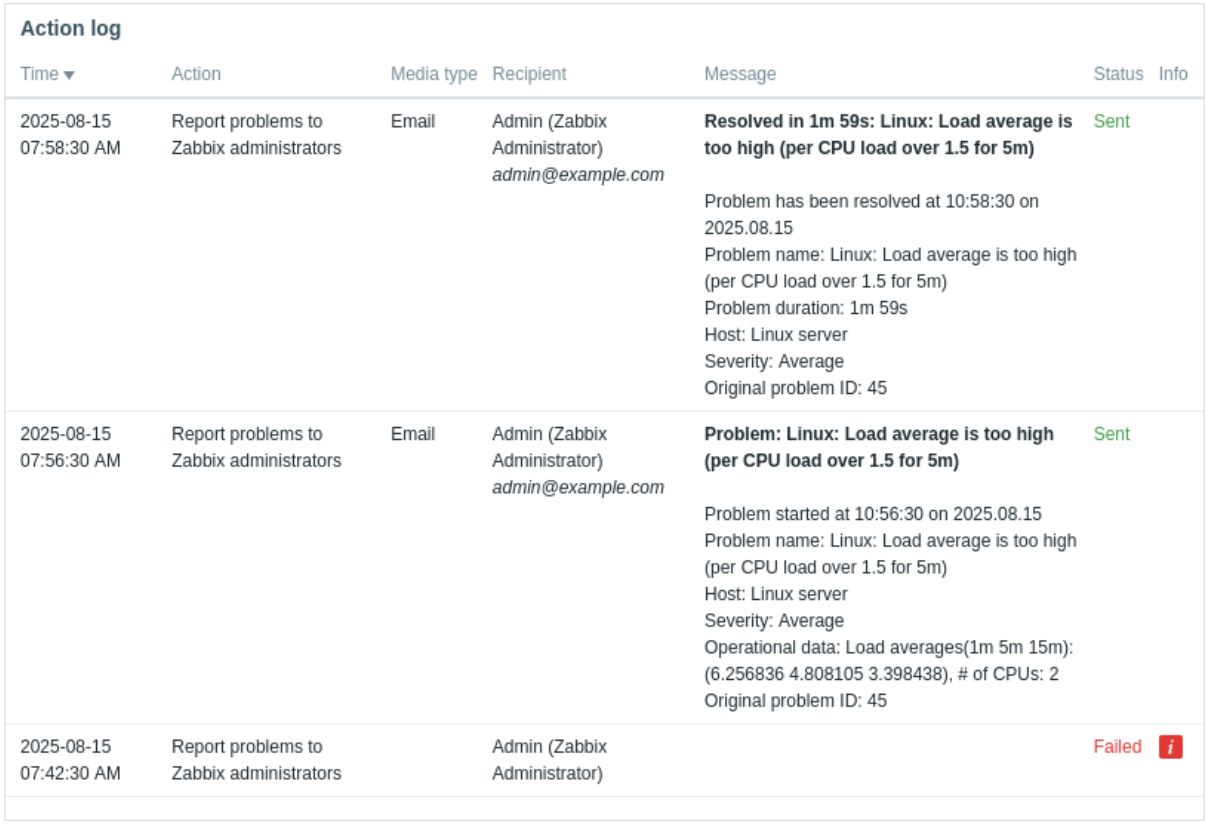
- Jeśli widżet będący źródłem danych nie rozgłasza danych, nasłuchujący widżet przechodzi w stan *Oczekiwanie na dane*.
- Jeśli widżet będący źródłem danych został usunięty, zastąpiony niekompatybilnym widżetem lub przeniesiony na inną stronę pulpitu, nasłuchujący widżet przechodzi w stan *Wskazany widżet jest niedostępny*.
- Jeśli określony host w źródle danych (widżecie lub pulpicie) nie ma encji skonfigurowanej w nasłuchującym widżecie (pozycji, wykresu, mapy itp.) lub jeśli użytkownik nie ma uprawnień dostępu do hosta, nasłuchujący widżet wyświetla następujący komunikat: *„Brak uprawnień do wskazanego obiektu lub obiekt nie istnieje!”*



Dziennik akcji

Przegląd

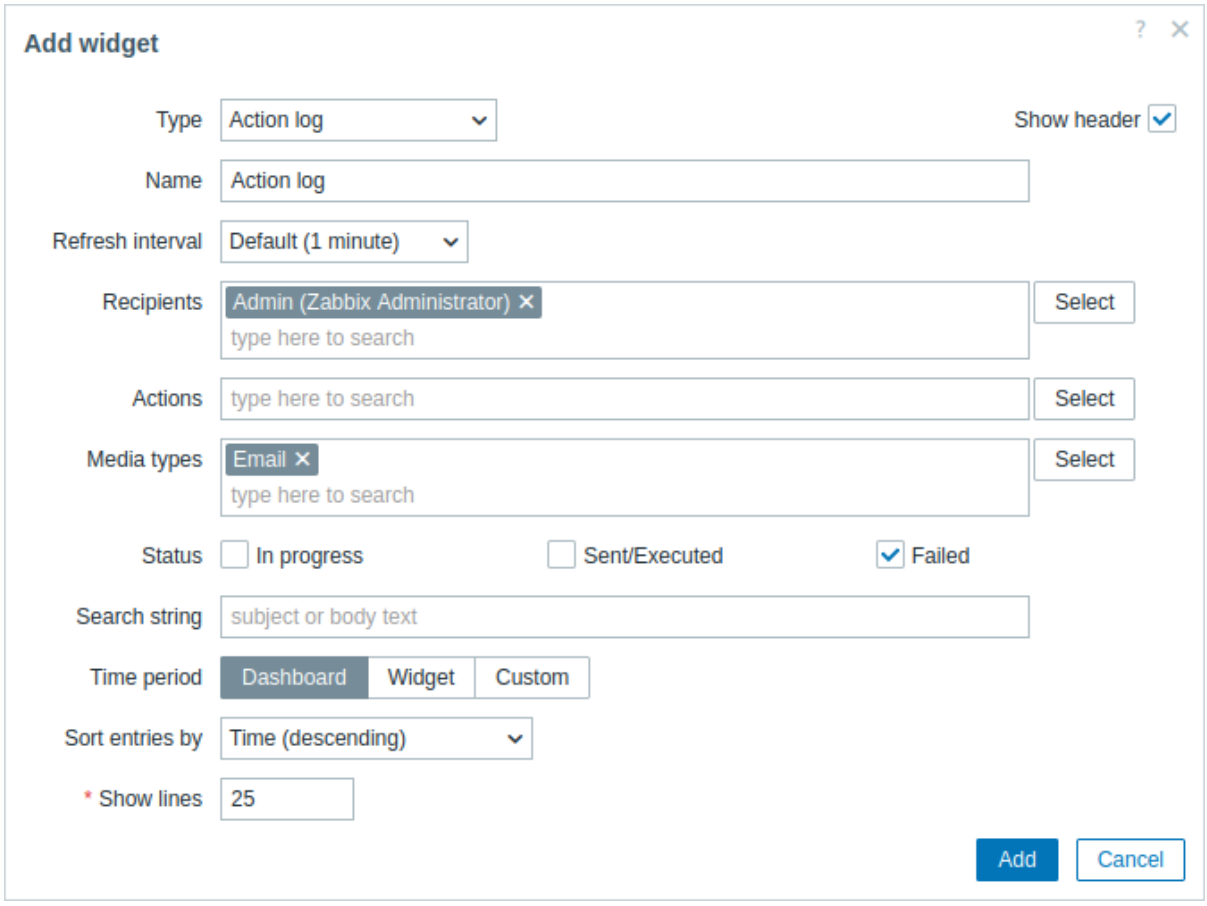
Widżet *Dziennik akcji* wyświetla szczegóły operacji (powiadomień, zdalnych poleceń) wykonanych w ramach akcji. Szczegółowy dziennik pomaga w audycie, rozwiązywaniu problemów i monitorowaniu wykonania każdej akcji.



Wyświetla te same dane co *Raporty* > *Dziennik akcji* i może wyświetlać do 1000 rekordów.

Konfiguracja

Aby skonfigurować widżet, wybierz *Action log* jako typ:



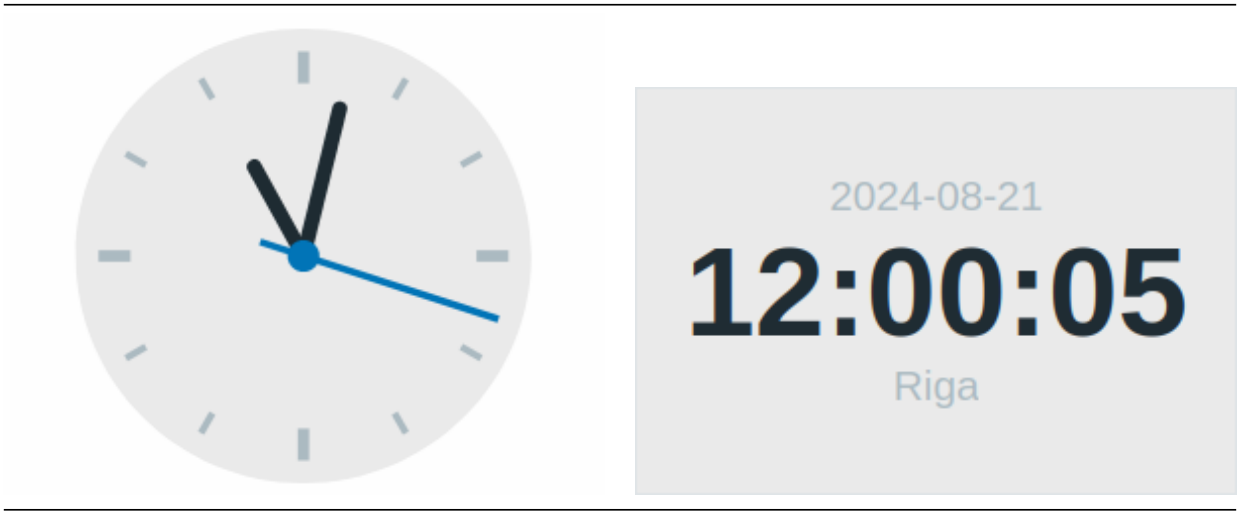
Oprócz parametrów wspólnych dla wszystkich widżetów, możesz ustawić następujące opcje specyficzne:

Recipients	Filtruj wpisy według odbiorców. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy odbiorcy zostanie wyświetlona rozwijana lista pasujących odbiorców. Jeśli nie wybrano żadnych odbiorców, zostaną wyświetlone szczegóły operacji akcji dla wszystkich odbiorców.
Actions	Filtruj wpisy według akcji. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy akcji zostanie wyświetlona rozwijana lista pasujących akcji. Jeśli nie wybrano żadnych akcji, zostaną wyświetlone szczegóły operacji akcji dla wszystkich akcji.
Media types	Filtruj wpisy według typów mediów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy typu mediów zostanie wyświetlona rozwijana lista pasujących typów mediów. Jeśli nie wybrano żadnych typów mediów, zostaną wyświetlone szczegóły operacji akcji dla wszystkich typów mediów.
Status	Zaznacz pole wyboru, aby filtrować wpisy według odpowiedniego statusu: **In progress** - wyświetlane są operacje akcji będące w toku; **Sent/Executed** - wyświetlane są operacje akcji, które wysłały powiadomienie lub zostały wykonane; **Failed** - wyświetlane są operacje akcji, które zakończyły się niepowodzeniem.
Search string	Filtruj wpisy według treści wiadomości/zdalnego polecenia. Jeśli wpiszesz tutaj ciąg znaków, wyświetlane będą tylko te operacje akcji, których wiadomość/zdalne polecenie zawiera podany ciąg. Makra nie są rozwijane.
Time period	Filtruj wpisy według okresu czasu. Wybierz źródło danych dla okresu czasu: **Dashboard** - użyj selektora okresu czasu pulpitu; **Widget** - użyj zgodnego widżetu (ustawionego w parametrze *Widget*); **Custom** - użyj niestandardowego okresu czasu ustawionego w parametrach *From* i *To*; jeśli zostanie ustawiony, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem. Zwróć uwagę, że niezależnie od konfiguracji *Time period* widżetu, zgodne widżety nadal mogą używać go jako źródła danych dla okresu czasu.
Widget	Wprowadź lub wybierz zgodny widżet jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Widget".
From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Custom".
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Custom".
Sort entries by	Sortuj wpisy według: **Time** (malejąco lub rosnąco); **Type** (malejąco lub rosnąco); **Status** (malejąco lub rosnąco); **Recipient** (malejąco lub rosnąco).
Show lines	Ustaw, ile wierszy dziennika akcji będzie wyświetlanych w widżecie.

Zegar

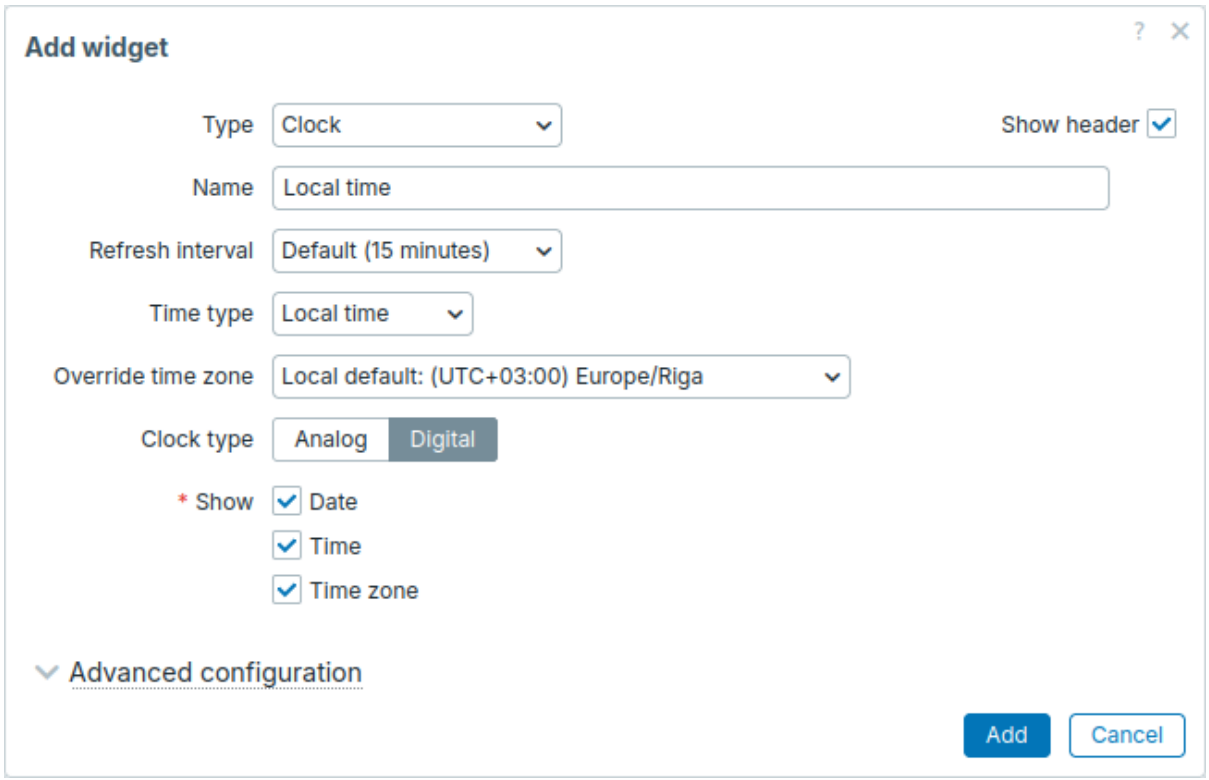
Przegląd

Widżet *Zegar* wyświetla czas lokalny, czas serwera lub hosta w formacie analogowym albo cyfrowym.



Konfiguracja

Aby skonfigurować widget, wybierz *Zegar* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów możesz ustawić następujące opcje:

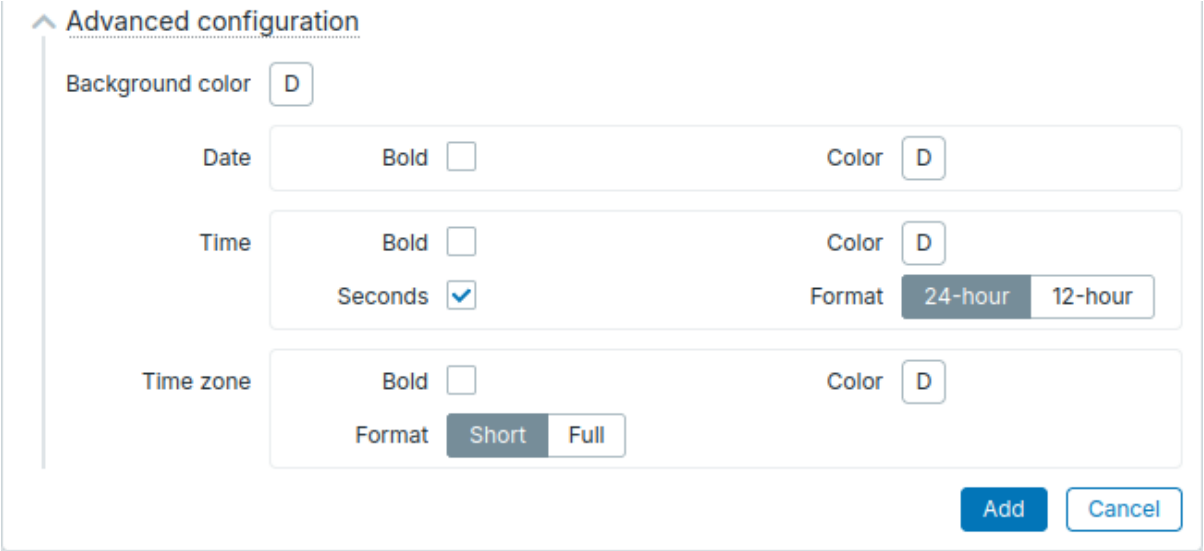
Typ czasu	Wybierz źródło czasu: **Czas lokalny** - czas systemowy urządzenia, na którym działa przeglądarka; **Czas serwera** - strefa czasowa ustawiona globalnie lub dla użytkownika Zabbix; **Czas hosta** - czas pobrany z pozycji hosta.
Pozycja	Wybierz pozycję z kluczem pozycji `system.localtime[local]`, aby wyświetlać czas hosta. Ten parametr jest dostępny, jeśli dla opcji *Typ czasu* ustawiono wartość „Czas hosta”.
Typ zegara	Wybierz typ zegara (analogowy lub cyfrowy).
Zastąp strefę czasową	Wybierz strefę czasową. Ten parametr jest dostępny dla typów czasu *Czas lokalny* i *Czas serwera*.
Pokaż	Wybierz co najmniej jedną jednostkę informacji (datę, czas, strefę czasową) do wyświetlenia na zegarze cyfrowym. Rozmiar jednostki jest automatycznie dostosowywany do wyświetlanych jednostek, rozmiaru widgetu i skalowania ekranu. Ten parametr jest dostępny, jeśli dla opcji *Typ zegara* ustawiono wartość „Cyfrowy”.

Konfiguracja zaawansowana	Kliknij nagłówek *Konfiguracja zaawansowana*, aby wyświetlić opcje konfiguracji zaawansowanej zegara cyfrowego. Ta sekcja jest dostępna, jeśli dla opcji *Typ zegara* ustawiono wartość „Cyfrowy”.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana* i tylko dla tych elementów, które zostały wybrane w polu *Pokaż* (zobacz powyżej).

Ponadto konfiguracja zaawansowana umożliwia zmianę koloru tła całego widżetu.

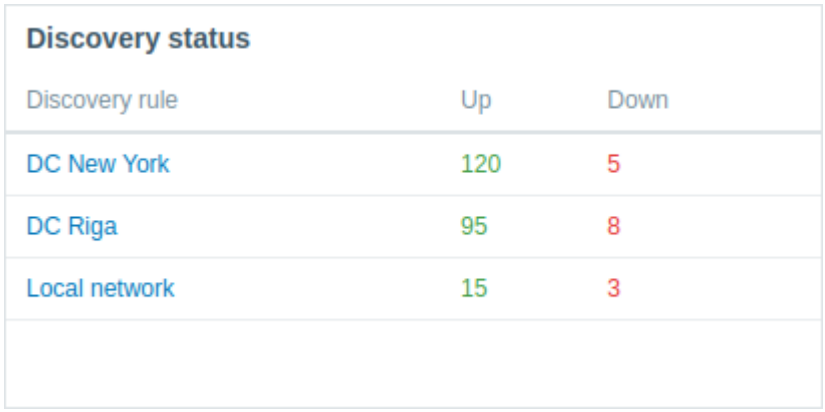


Kolor tła		Wybierz kolor tła z selektora kolorów. D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.
Data		Ta sekcja jest dostępna, gdy w parametrze *Pokaż* zaznaczono pole wyboru *Data*.
	Pogrubienie	Zaznacz pole wyboru, aby wyświetlać datę pogrubioną czcionką.
	Kolor	Wybierz kolor daty z selektora kolorów. D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.
Czas		Ta sekcja jest dostępna, gdy w parametrze *Pokaż* zaznaczono pole wyboru *Czas*.
	Pogrubienie	Zaznacz pole wyboru, aby wyświetlać czas pogrubioną czcionką.
	Kolor	Wybierz kolor czasu z selektora kolorów. D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.
	Sekundy	Zaznacz pole wyboru, aby wyświetlać sekundy. W przeciwnym razie będą wyświetlane tylko godziny i minuty.
	Format	Wybierz wyświetlanie czasu w formacie 24- lub 12-godzinnym.
Strefa czasowa		Ta sekcja jest dostępna, gdy w parametrze *Pokaż* zaznaczono pole wyboru *Strefa czasowa*.
	Pogrubienie	Zaznacz pole wyboru, aby wyświetlać strefę czasową pogrubioną czcionką.
	Kolor	Wybierz kolor strefy czasowej z selektora kolorów. D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.
	Format	Wybierz wyświetlanie strefy czasowej w krótkim formacie (np. `New York`) lub pełnym formacie (np.,`(UTC-04:00) America/New York`).

Status wykrywania

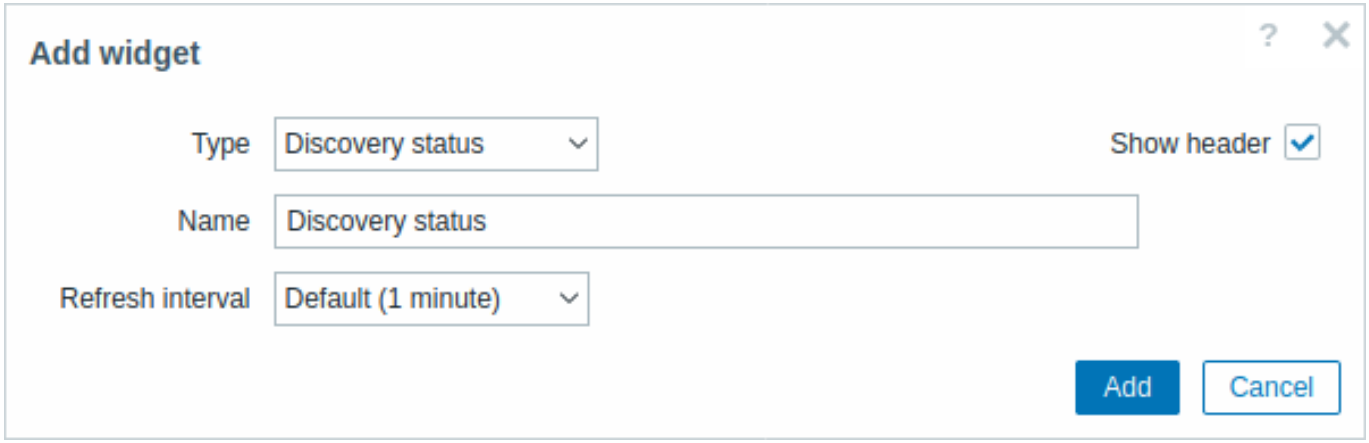
Przegląd

Widżet *Status wykrywania* wyświetla status urządzeń wykrytych przez włączone reguły wykrywania sieci. Pomaga monitorować, czy wykrywanie sieci działa prawidłowo, oraz liczbę urządzeń, które są dostępne lub niedostępne.



Konfiguracja

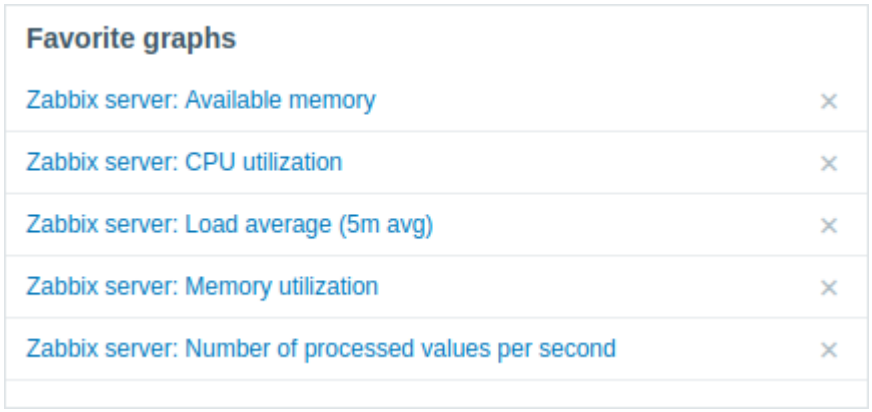
Wszystkie parametry konfiguracji dla tego widżetu są wspólne dla wszystkich widżetów.



Ulubione wykresy

Przegląd

Widżet *Ulubione wykresy* wyświetla alfabetyczną listę skrótów do wykresów oznaczonych jako ulubione przez bieżącego użytkownika.



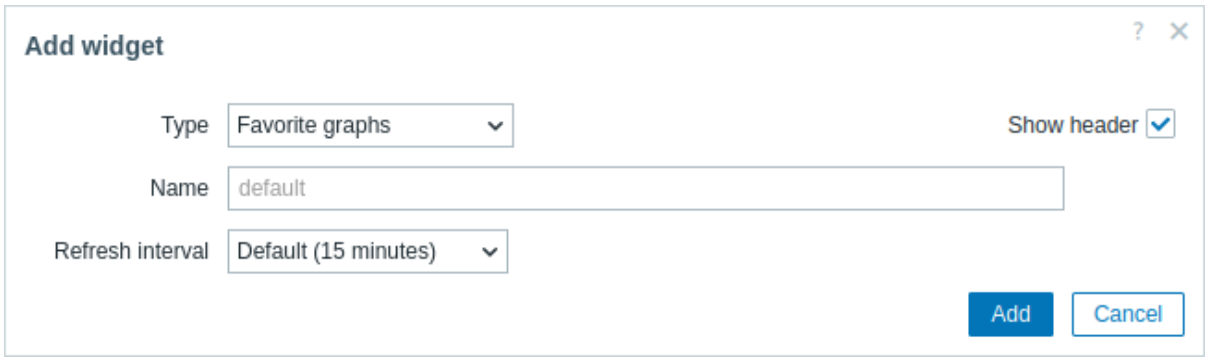
Kliknięcie x usuwa wykres z listy oraz z ulubionych.

Wykresy są oznaczane jako ulubione po kliknięciu przycisku  *Dodaj do ulubionych* podczas przeglądania wykresów pozycji w *Monitoring* > *Najnowsze dane*.



Konfiguracja

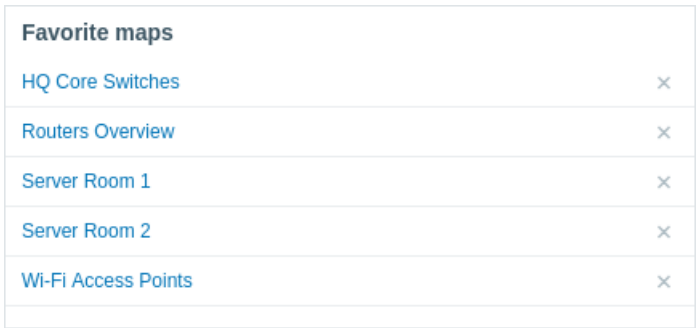
Wszystkie parametry konfiguracji tego widżetu są wspólne dla wszystkich widżetów.



Ulubione mapy

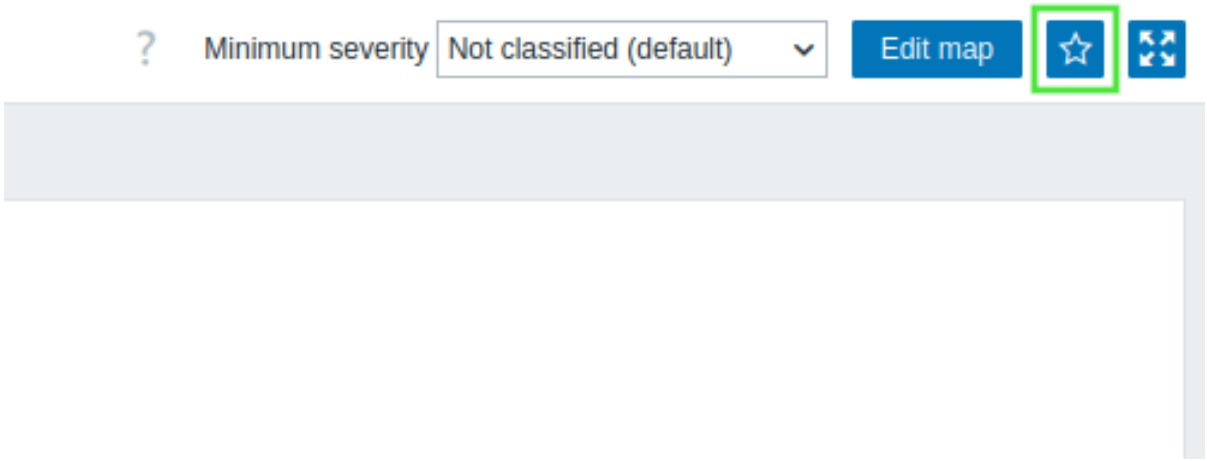
Przegląd

Widżet *Ulubione mapy* wyświetla listę skrótów do map oznaczonych przez bieżącego użytkownika jako ulubione, posortowaną alfabetycznie.



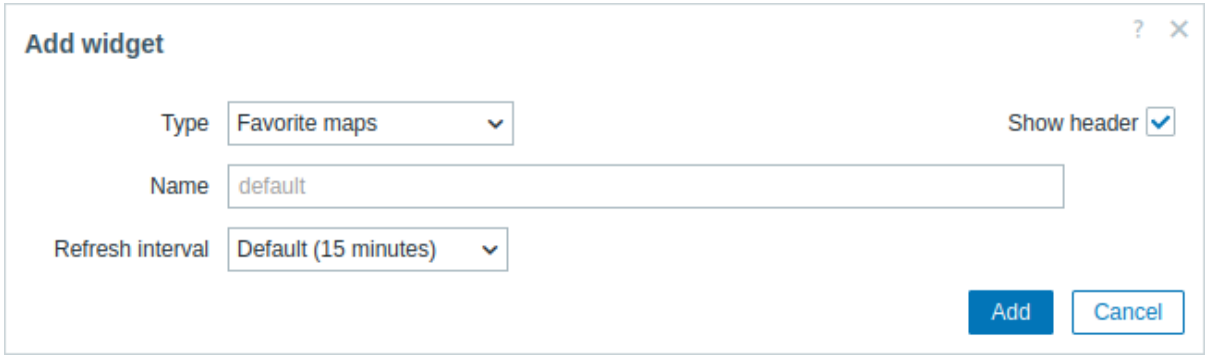
Kliknięcie x usuwa mapę z ulubionych.

Mapy są oznaczane jako ulubione po kliknięciu przycisku  *Dodaj do ulubionych* podczas przeglądania map w *Monitoring* > *Mapy*.



Konfiguracja

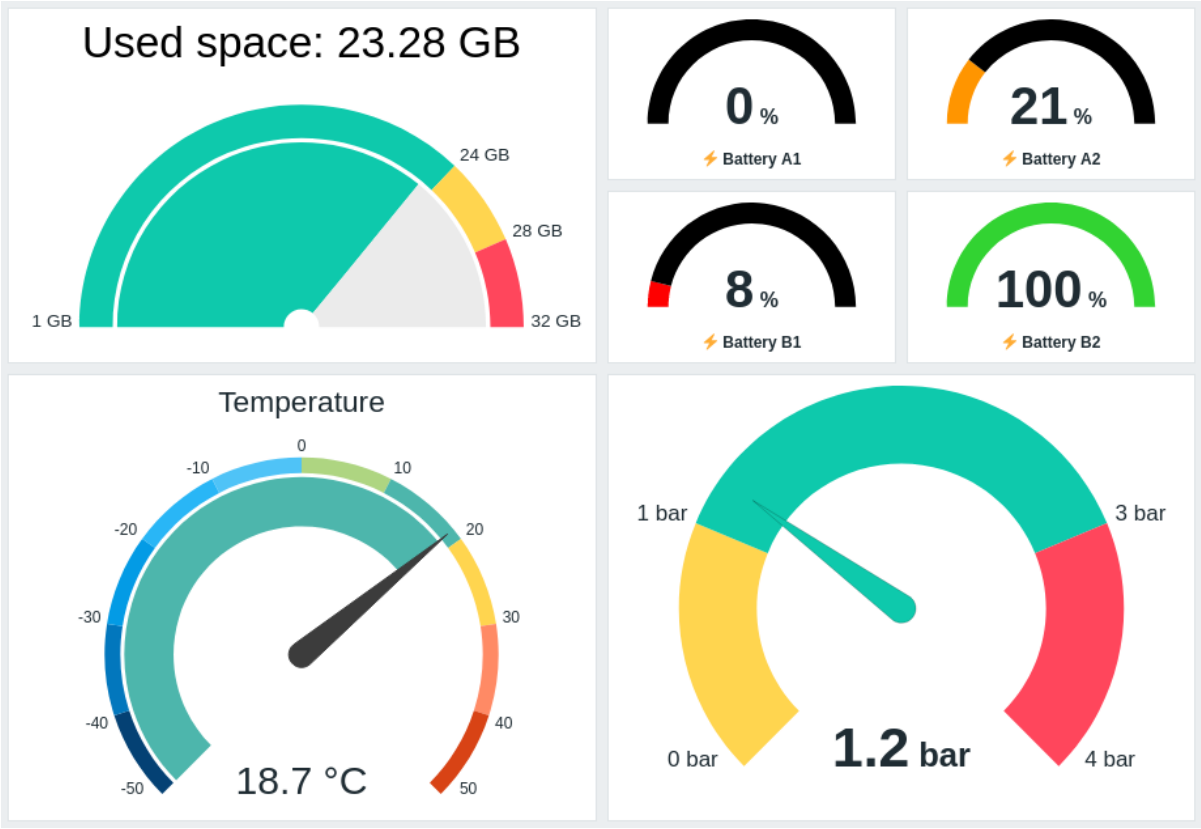
Wszystkie parametry konfiguracji tego widżetu są wspólne dla wszystkich widżetów.



Wskaźnik

Przegląd

Widżet *Gauge* wyświetla wartość pojedynczej numerycznej pozycji w formie wskaźnika. Jest przydatny do monitorowania ważnych metryk, wizualizacji progów i wykrywania nagłych zmian.



Widżet można skonfigurować tak, aby wyświetlał:

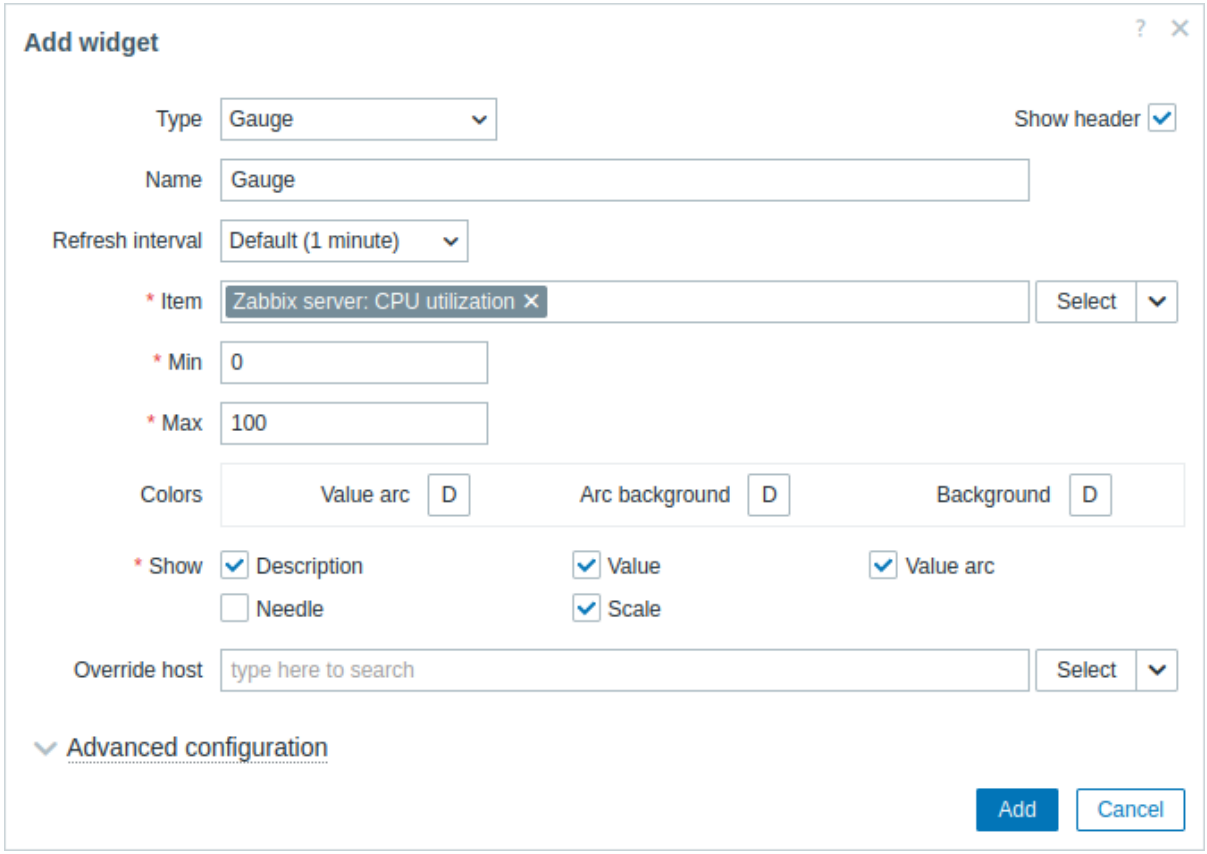
- Opis pozycji (Wykorzystane miejsce: {ITEM.LASTVALUE}, Temperatura)
- Wartość pozycji i jednostki (21 %, 18.7 °C, 1.2 bar)
- Skalę wskaźnika (1GB/24GB/28GB/32GB, -50/-40/-30/itp.)
- Łuk wskaźnika (łuk wartości wskaźnika i łuk progów wskaźnika)
- Wskazówkę wskaźnika

Kliknięcie widżetu otwiera prosty wykres dla pozycji.

Informacje wyświetlane w widżecie *Gauge* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu *Działania* widżetu …, które pojawia się po umieszczeniu kursora nad widżetem. W zależności od ustawień przeglądarki obraz zostanie automatycznie zapisany w folderze `Downloads` albo otworzy się okno dialogowe *Zapisz*, umożliwiające wybranie lokalizacji.

Konfiguracja

Aby skonfigurować, wybierz *Gauge* jako typ:

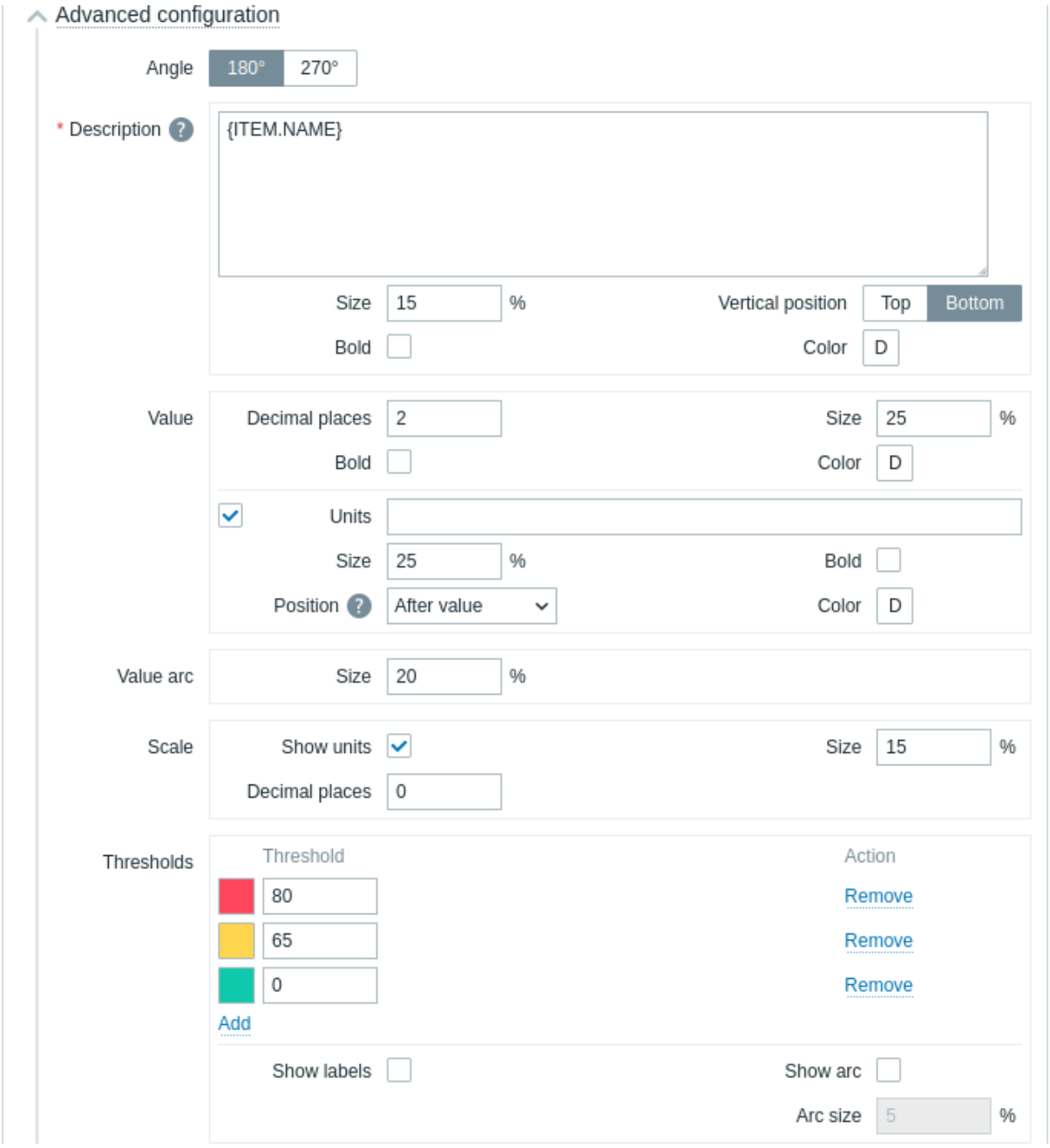


Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Item	Wybierz pozycję. Alternatywnie wybierz kompatybilny widżet jako źródło danych dla pozycji. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy pozycji zostanie wyświetlona rozwijana lista pasujących pozycji. Pamiętaj, że możesz wybierać tylko pozycje zwracające dane numeryczne (z wyjątkiem danych binarnych).
Min	Wprowadź minimalną wartość wskaźnika. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.
Max	Wprowadź maksymalną wartość wskaźnika. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.
Colors	Wybierz kolor z próbnika kolorów: **Value arc** — wybierz kolor łuku wartości wskaźnika; **Arc background** — wybierz kolor tła łuku wartości wskaźnika i łuku progów wskaźnika; **Background** — wybierz kolor tła widżetu. „D” oznacza kolor domyślny, który zależy od motywu frontend. Jeśli ustawiono *Thresholds*, domyślny kolor dla **Value arc** zależy od koloru progu. Aby powrócić do koloru domyślnego, kliknij przycisk *Use default* w próbniku kolorów.
Show	Zaznacz pole wyboru, aby wyświetlić odpowiedni element wskaźnika — opis, wartość, łuk wartości, wskazówkę, skalę (minimalną i maksymalną wartość wskaźnika na początku i końcu łuku wskaźnika). Usuń zaznaczenie, aby ukryć. Musi być wybrany co najmniej jeden element. Pamiętaj, że wskazówka i skala wskaźnika mogą być wyświetlane, jeśli wyświetlany jest łuk wartości wskaźnika lub łuk progów wskaźnika (zobacz opcje konfiguracji zaawansowanej). Zwróć też uwagę, że jeśli wskazówka wskaźnika jest wyświetlana, wartość jest umieszczana pod wskazówką; jeśli wskazówka jest ukryta, wartość jest wyrównana do dolnej części łuku wskaźnika.
Override host	Wybierz kompatybilny widżet lub dashboardowy selektor hosta jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje konfiguracji zaawansowanej. Tutaj będzie też można dostosować elementy wskaźnika wybrane w polu *Show*.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*:



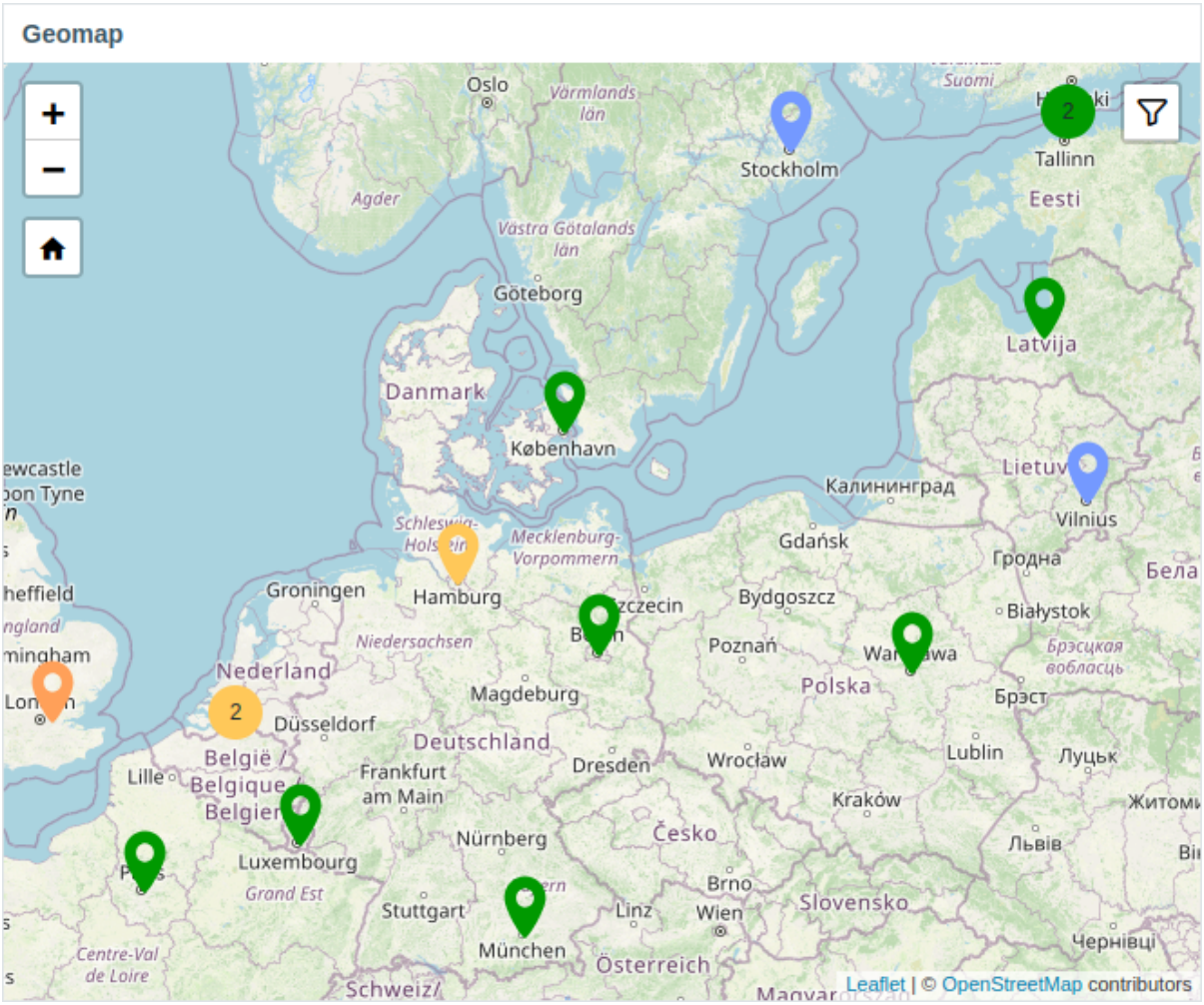
Kąt	Wybierz kąt wskaźnika (180° lub 270°).
Opis	
Opis	Wprowadź opis pozycji. Ten opis może zastąpić domyślną nazwę pozycji. Obsługiwane są opisy wielowierszowe. Możliwe jest połączenie tekstu i obsługiwanych makr. Obsługiwane są {HOST.*}, {ITEM.*}, {INVENTORY.*} oraz makra użytkownika.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla opisu pozycji (w procentach, względem całkowitej wysokości widżetu).
Pozycja pionowa	Wybierz pionowe położenie opisu pozycji (u góry lub u dołu, względem łuku wskaźnika).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać opis pozycji pogrubioną czcionką.
Kolor	Wybierz kolor opisu pozycji z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Wartość	
Miejsca dziesiętne	Wprowadź liczbę miejsc dziesiętnych wyświetlanych przy wartości. Ta opcja ma wpływ tylko na pozycje zwracające dane numeryczne (float).
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla wartości (w procentach, względem wysokości łuku wskaźnika).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać wartość pogrubioną czcionką.

Kolor	Wybierz kolor wartości z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Jednostki	
Jednostki	Zaznacz pole wyboru, aby wyświetlać jednostki wraz z wartością pozycji. Jeśli wprowadzisz nazwę jednostki, zastąpi ona jednostki ustawione w konfiguracji pozycji.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla jednostek pozycji (w procentach, względem wysokości łuku wskaźnika).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać jednostki pozycji pogrubioną czcionką.
Pozycja	Wybierz położenie jednostek pozycji (nad, pod, przed lub za, względem wartości pozycji). Ta opcja jest ignorowana dla następujących jednostek związanych z czasem: unixtime, uptime, s.
Kolor	Wybierz kolor jednostek pozycji z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Łuk wartości	
Rozmiar łuku	Wprowadź wysokość rozmiaru łuku wartości wskaźnika (w procentach, względem promienia łuku wskaźnika).
Wskazówka	
Kolor	Wybierz kolor wskazówki wskaźnika z próbnika kolorów. ”D” oznacza kolor domyślny, który zależy od motywu frontend. Jeśli ustawiono *Progi*, domyślny kolor wskazówki zależy od koloru progu. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Skala	
Pokaż jednostki	Zaznacz pole wyboru, aby wyświetlać jednostki przy minimalnej i maksymalnej wartości wskaźnika.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla minimalnej i maksymalnej wartości wskaźnika (w procentach, względem wysokości łuku wskaźnika).
Miejsca dziesiętne	Wprowadź liczbę miejsc dziesiętnych wyświetlanych przy minimalnej i maksymalnej wartości wskaźnika. Ta opcja ma wpływ tylko na pozycje zwracające dane numeryczne (float).
Progi	
Progi	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu z próbnika kolorów i określ wartość numeryczną. Lista progów zostanie posortowana rosnąco podczas zapisywania. Należy pamiętać, że kolory skonfigurowane jako progi będą wyświetlane poprawnie tylko dla pozycji numerycznych. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.
Pokaż etykiety	Zaznacz pole wyboru, aby wyświetlać wartości progów jako etykiety na skali wskaźnika.
Pokaż łuk	Zaznacz pole wyboru, aby wyświetlać łuk progów wskaźnika.
Rozmiar łuku	Wprowadź wysokość rozmiaru łuku progów wskaźnika (w procentach, względem promienia łuku wskaźnika).

Geomap

Przegląd

Widżet *Geomap* wyświetla hosty jako znaczniki na interaktywnej mapie opartej na Leaflet, a kolory znaczników wskazują najpoważniejszy problem każdego hosta.

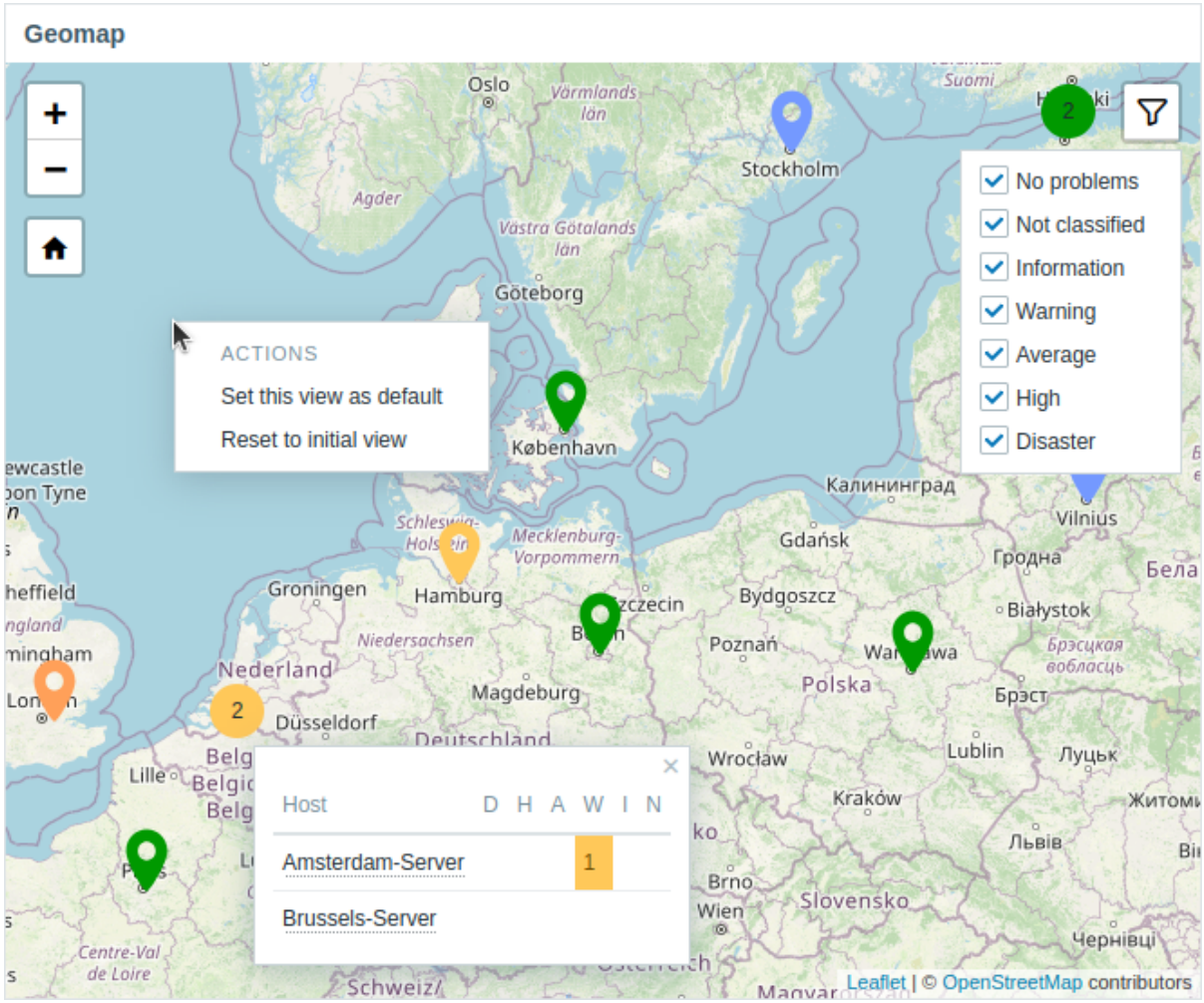


Note:
Zabbix oferuje wiele predefiniowanych dostawców kafelków mapy i umożliwia dodawanie własnych w *Administracja* > *Ogólne* > *Mapy geograficzne*.

Domyślnie widżet wyświetla wszystkie włączone hosty, które mają prawidłowe współrzędne (szerokość geograficzna od -90 do 90, długość geograficzna od -180 do 180) w swoim inwentarzu.

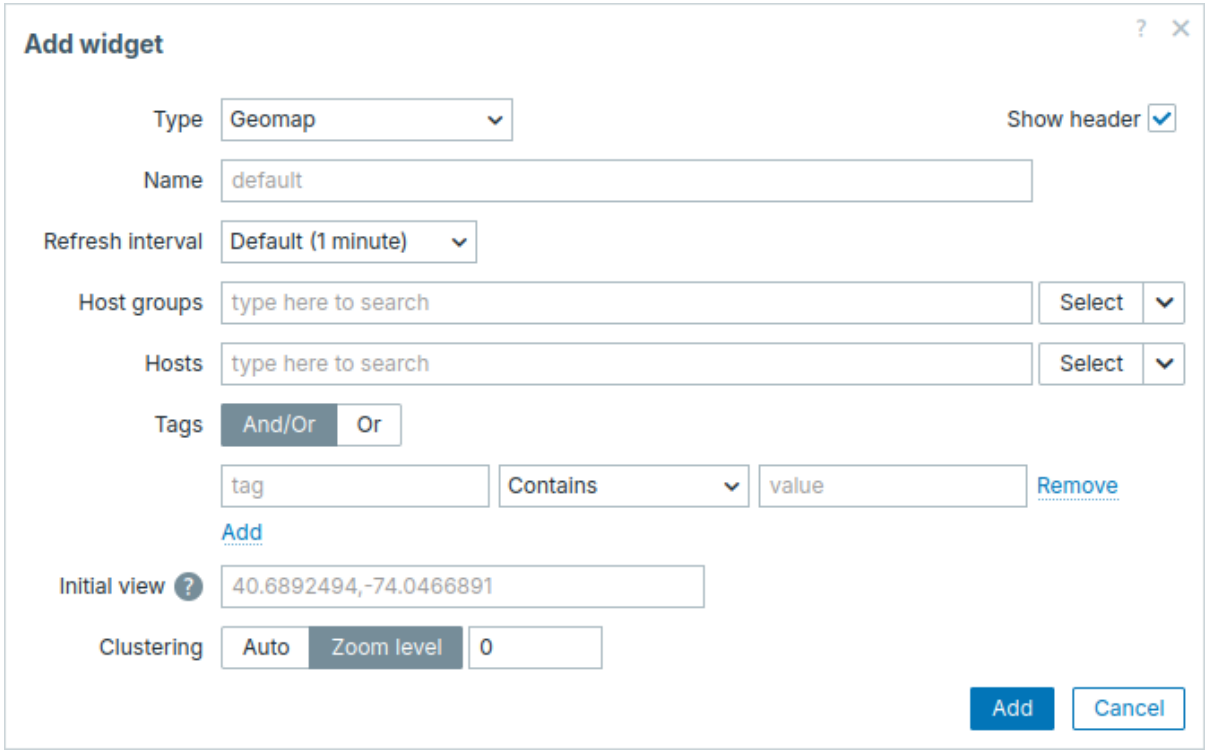
Użyj następujących opcji, aby korzystać z widżetu:

- Kliknięcie znacznika powoduje wyświetlenie nazwy hosta i nierozwiązanych problemów; kliknięcie nazwy hosta otwiera menu hosta.
- Filtruj hosty według ważności problemu za pomocą ikony filtra w prawym górnym rogu.
- Kliknij mapę prawym przyciskiem myszy, aby ustawić lub zresetować widok domyślny; użyj ikony domu, aby wrócić do widoku domyślnego.



Konfiguracja

Aby dodać widget, wybierz *Geomap* jako typ.



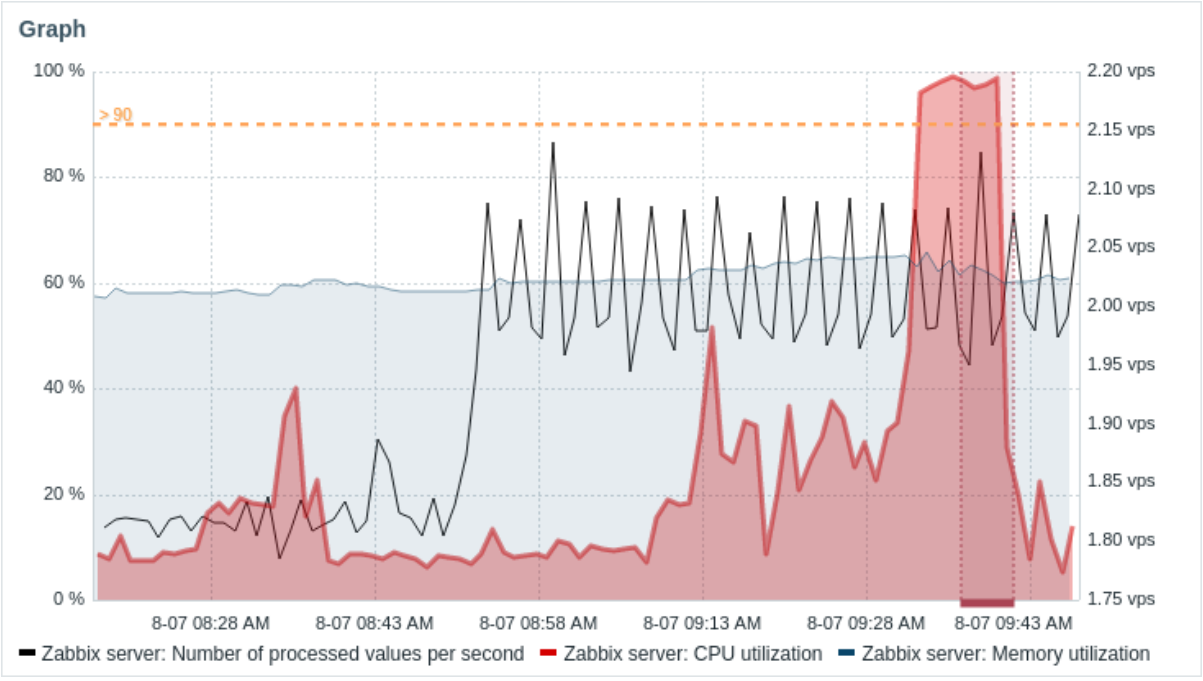
Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Grupy hostów	Wybierz grupy hostów, które mają być wyświetlane na mapie. Alternatywnie wybierz kompatybilny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Jeśli w polach *Grupy hostów* i *Hosty* nic nie zostanie wybrane, zostaną wyświetlone wszystkie hosty z prawidłowymi współrzędnymi. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Hosty	Wybierz hosty, które mają być wyświetlane na mapie. Alternatywnie wybierz kompatybilny widget lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli w polach *Grupy hostów* i *Hosty* nic nie zostanie wybrane, zostaną wyświetlone wszystkie hosty z prawidłowymi współrzędnymi. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Tagi	Określ tagi, aby ograniczyć liczbę hostów wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane warunkiem *Or*; **Or** - wystarczy spełnienie jednego warunku. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Widok początkowy	Oddzielone przecinkami współrzędne środka oraz opcjonalny poziom powiększenia, które mają zostać wyświetlone przy początkowym załadowaniu widgetu, w formacie `<latitude>,<longitude>,<zoom>` Jeśli początkowy poziom powiększenia jest określony, widget Geomap zostanie załadowany z podanym poziomem powiększenia. W przeciwnym razie początkowy poziom powiększenia jest obliczany jako połowa maksymalnego poziomu powiększenia dla danego dostawcy kafelków. Widok początkowy jest ignorowany, jeśli ustawiono widok domyślny (zobacz poniżej). Przykłady: 40.6892494,-74.0466891 40.6892494,-122.0466891
Klastrowanie	Określ, w jaki sposób znaczniki pobliskich hostów są łączone w jeden znacznik z licznikiem: **Auto** - znaczniki są łączone automatycznie; **Zoom level** - znaczniki są łączone na podstawie określonego poziomu powiększenia (0 - maksymalne oddalenie, 30 - maksymalne przybliżenie). Na przykład, jeśli parametr jest ustawiony na 0, znaczniki nie są łączone na żadnym poziomie powiększenia; jeśli ustawiono 15, znaczniki są łączone od maksymalnego oddalenia do poziomu powiększenia 15; jeśli ustawiono 30, znaczniki są łączone na wszystkich poziomach powiększenia. Zwróć uwagę, że maksymalny poziom przybliżenia jest określany przez dostawcę kafelków mapy.

Wykres

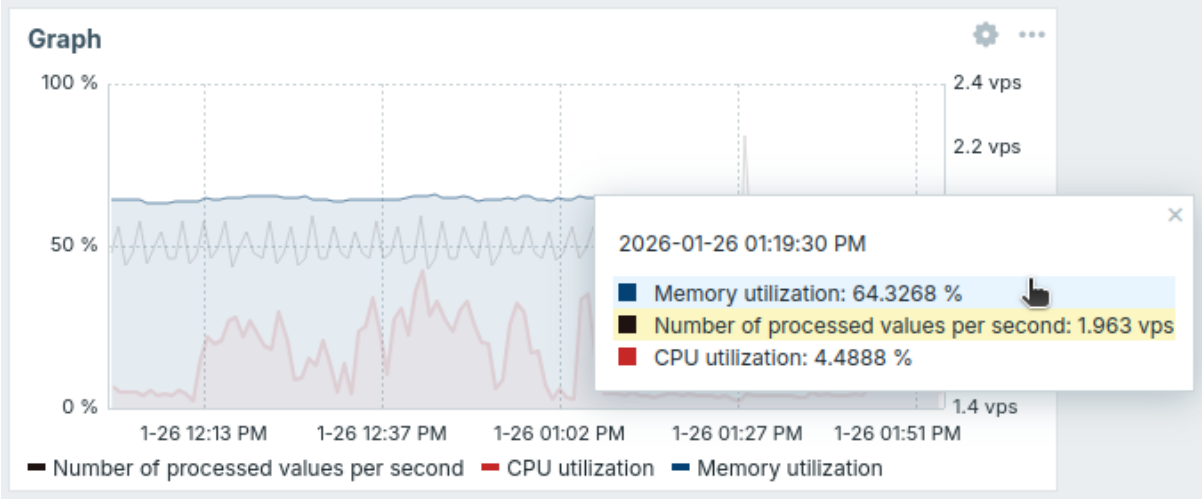
Przegląd

Widżet *Graph* wyświetla dane liczbowe z pozycji w postaci wykresu wektorowego. Może pomóc w śledzeniu trendów, wykrywaniu problemów oraz porównywaniu wartości w czasie i między hostami.

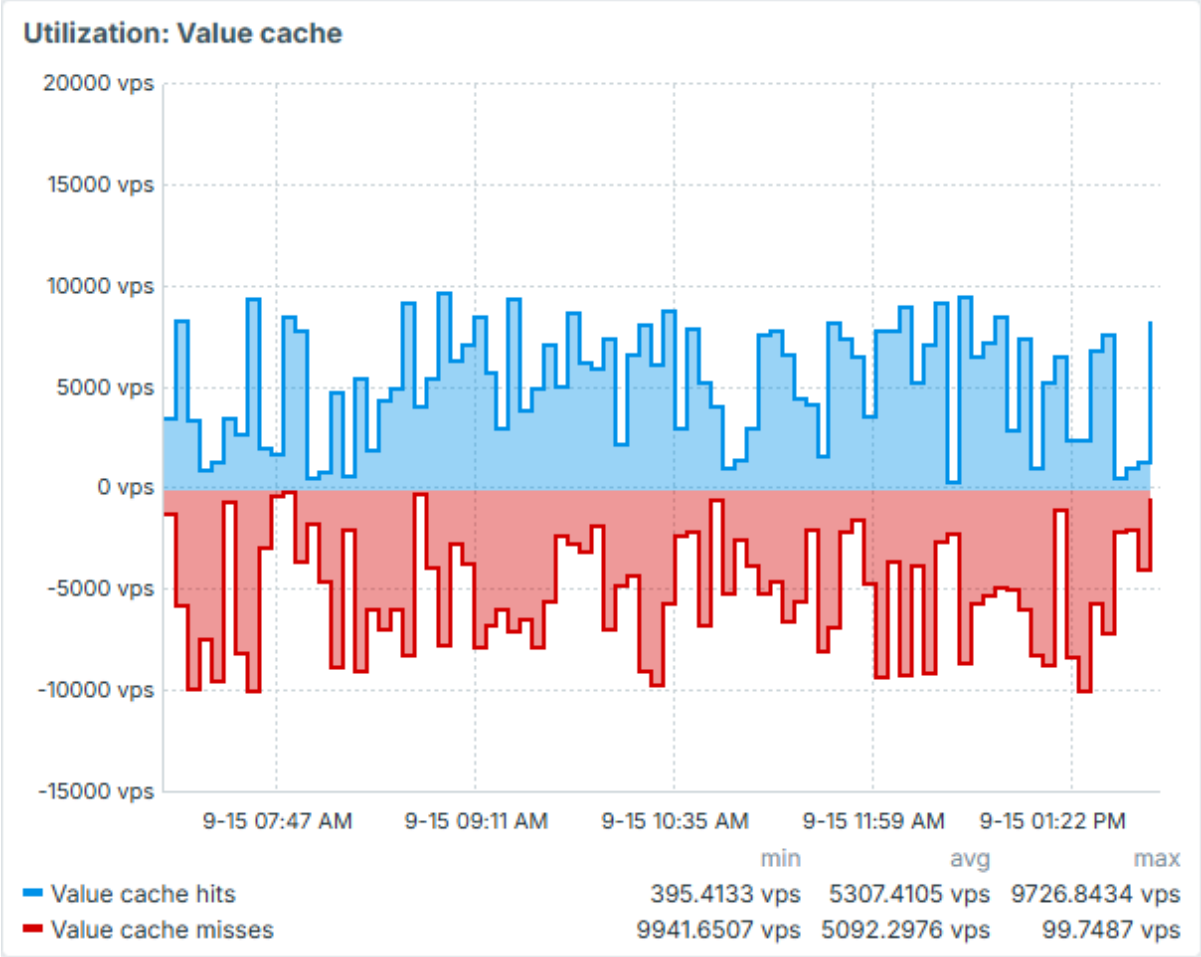


Po umieszczeniu wskaźnika myszy nad wykresem wyświetlana jest etykieta narzędzia z listą pozycji posortowanych według wartości w kolejności malejącej.

Po umieszczeniu wskaźnika myszy nad pozycją w etykiecie narzędzia jej wykres zostaje wyróżniony (pozostałe są przyciemnione); wybranie pozycji przekazuje jej dane do innych widżetów.



Wykres może być wyświetlany z odwróconymi wartościami osi Y; wpływa to wyłącznie na sposób prezentacji i nie zmienia oryginalnych wartości.

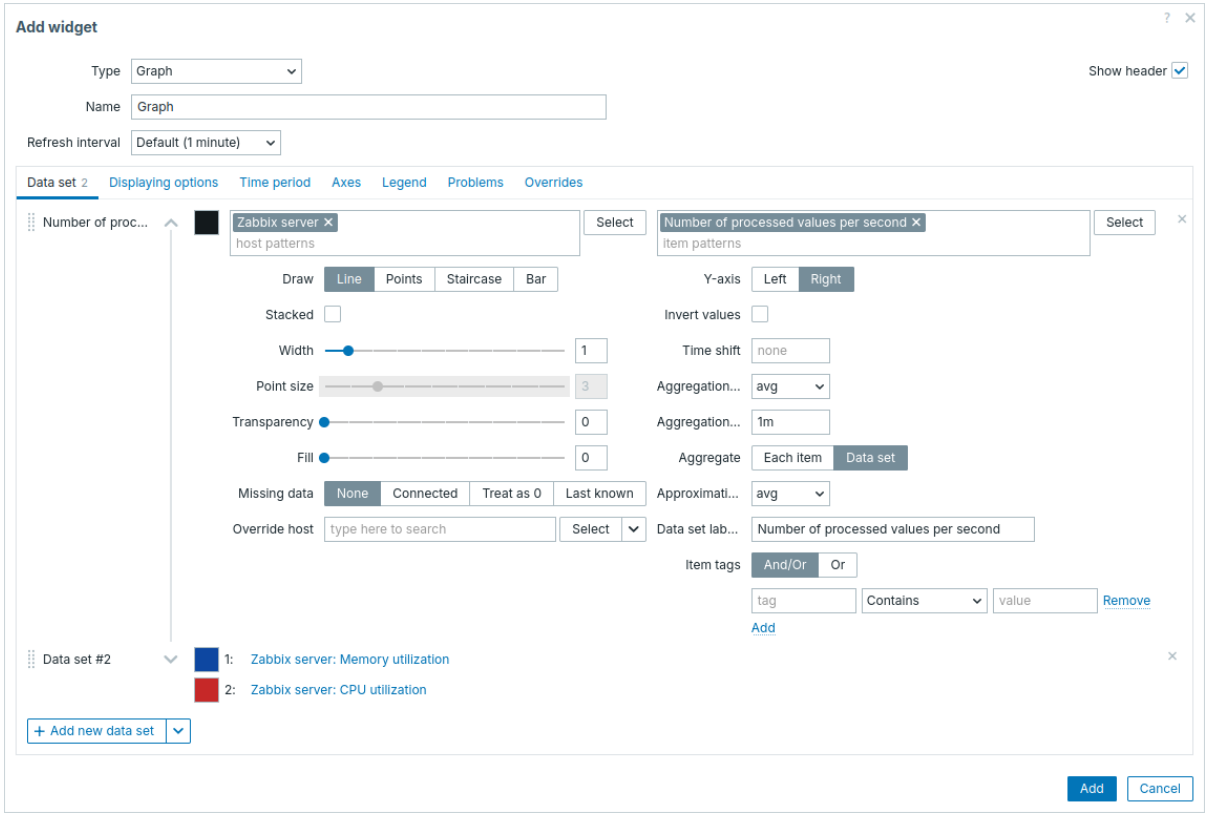


Dostępny jest również starszy widżet Graph (classic).

Informacje wyświetlane w widżecie *Graph* można pobrać jako obraz PNG, wybierając opcję *Download image* w menu *Actions* widżetu ⋮, które pojawia się po umieszczeniu wskaźnika myszy nad widżetem. W zależności od ustawień przeglądarki obraz jest automatycznie zapisywany w folderze `Downloads` albo otwierane jest okno dialogowe *Save*, umożliwiające wybranie lokalizacji.

Konfiguracja

Aby skonfigurować, wybierz *Graph* jako typ:



Zestaw danych

Karta **Zestaw danych** umożliwia wybór danych do wykresu poprzez dodawanie zestawów danych. Można dodać dwa typy zestawów danych:

- *Wzorce pozycji* - wyświetlane są dane z pasujących pozycji. Można wybrać pojedynczy kolor bazowy lub wiersz palety, aby przypisać różne kolory do każdej pasującej pozycji.
- *Lista pozycji* - wyświetlane są dane z wybranych pozycji. Kolor każdej pozycji można wybrać indywidualnie z selektora.

Domyślnie dodawany jest zestaw danych typu *Wzorce pozycji*.

Zestaw danych	Dla zestawu danych **Wzorce pozycji**: Wybierz lub wprowadź wzorce hostów i pozycji; dane pozycji pasujących do tych wzorców zostaną wyświetlone na wykresie; można wyświetlić maksymalnie 50 pozycji. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub większej liczby znaków). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij klawisz *Enter*. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać indywidualnie na przykład pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (na przykład *item2*, *item3*). Określenie wzorców hostów i pozycji jest wymagane dla zestawów danych typu „Wzorce pozycji”. Zobacz także: Szczegóły konfiguracji zestawu danych. Dla zestawu danych **Lista pozycji**: Wybierz pozycje do wyświetlenia na wykresie, klikając przycisk *Dodaj pozycję*. Można również wybrać kompatybilne widgety jako źródło danych dla pozycji, klikając przycisk *Dodaj widget*. Określenie pozycji lub widgetów jest wymagane dla zestawów danych typu „Lista pozycji”. Zobacz także: Szczegóły konfiguracji zestawu danych. Należy pamiętać, że dozwolone są tylko pozycje numeryczne. Podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu parametr określania wzorców hostów jest niedostępny, a parametr określania listy pozycji umożliwia wyłącznie wybór pozycji skonfigurowanych w szablonie.

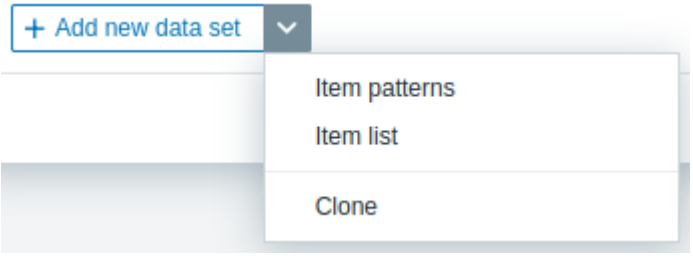
Rysowanie	Wybierz typ rysowania metryki. Dostępne typy rysowania: *Linia* (ustawiony domyślnie), *Punkty*, *Schodki* i *Słupki*. Należy pamiętać, że jeśli na wykresie liniowym/schodkowym znajduje się tylko jeden punkt danych, jest on rysowany jako punkt niezależnie od wybranego typu rysowania. Rozmiar punktu jest obliczany na podstawie szerokości linii, ale nie może być mniejszy niż 3 piksele, nawet jeśli szerokość linii jest mniejsza.
Skumulowany	Zaznacz pole wyboru, aby wyświetlać dane jako skumulowane (wyświetlane są wypełnione obszary). Ta opcja jest wyłączona po wybraniu typu rysowania *Punkty*.
Szerokość	Ustaw szerokość linii. Ta opcja jest dostępna po wybraniu typu rysowania *Linia* lub *Schodki*.
Rozmiar punktu	Ustaw rozmiar punktu. Ta opcja jest dostępna po wybraniu typu rysowania *Punkty*.
Przezroczystość	Ustaw poziom przezroczystości.
Wypełnienie	Ustaw poziom wypełnienia. Ta opcja jest dostępna po wybraniu typu rysowania *Linia* lub *Schodki*.
Brakujące dane	Wybierz opcję wyświetlania brakujących danych: **Brak** - luka pozostaje pusta; **Połączone** - łączone są dwie wartości graniczne; **Traktuj jako 0** - brakujące dane są wyświetlane jako wartości 0; **Ostatnia znana** - brakujące dane są wyświetlane z taką samą wartością jak ostatnia znana wartość; nie dotyczy typów rysowania *Punkty* i *Słupki*.
Zastąp host	Wybierz kompatybilny widget lub selektor hostów pulpitu nawigacyjnego jako źródło danych dla hostów. Ten parametr jest niedostępny podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu.
Oś Y	Wybierz stronę wykresu, po której będzie wyświetlana oś Y.
Odwróć wartości	Zaznacz pole wyboru, aby przed wyświetleniem na wykresie pomnożyć wszystkie wartości osi Y przez -1. Wpływa to wyłącznie na reprezentację wizualną; oryginalne dane pozostają niezmienione. Ta opcja nie wpływa na etykietę narzędzia, legendę ani linię percentyla. Jeśli dla osi wybrano opcję *Proste wyzwalacze*, progi wyzwalaczy zostaną wyświetlone w odwrócony sposób.
Przesunięcie czasu	W razie potrzeby określ przesunięcie czasu. W tym polu można używać przyrostków czasu. Dozwolone są wartości ujemne.
Funkcja agregacji	Określ funkcję agregacji używaną dla każdej pozycji lub całego zestawu danych w ramach wybranego *Interwału agregacji*: **min** - wyświetla najmniejszą wartość; **max** - wyświetla największą wartość; **avg** - wyświetla wartość średnią; **sum** - wyświetla sumę wartości; **count** - wyświetla liczbę wartości; **first** - wyświetla pierwszą wartość; **last** - wyświetla ostatnią wartość. Jeśli wybrano **nieużywana**, wyświetlane są wszystkie wartości (bez agregacji). Dla funkcji *min*, *max*, *avg* i *sum* można wyświetlać tylko dane numeryczne. W przypadku funkcji *count* dane nienumeryczne zostaną zmienione na numeryczne. Zobacz także: Agregacja na wykresach.
Interwał agregacji	Określ interwał agregowania wartości. W tym polu można używać przyrostków czasu. Wartość liczbowa bez przyrostka będzie traktowana jako liczba sekund. Należy pamiętać, że jeśli widget jest skonfigurowany do wyświetlania danych historycznych na podstawie trendów (*Wybór danych historycznych* ustawiono na *Trendy* lub *Automatycznie*), zaleca się użycie interwału agregacji będącego wielokrotnością 1 godziny (np. 3600, 60m, 1h, 3h itd.). Trendy przechowują zagregowane wartości godzinowe, dlatego użycie interwału agregacji, który nie jest wielokrotnością 1 godziny (np. 100s, 7min, 15min, 90min itd.), może prowadzić do wyników trudnych do zinterpretowania.

Agreguj	Określ, czy agregować: **Każdą pozycję** - każda pozycja w zestawie danych zostanie zagregowana i wyświetlona osobno; **Zestaw danych** - wszystkie pozycje zestawu danych zostaną zagregowane i wyświetlone jako jedna wartość.
Przybliżenie	Określ, jaką wartość wyświetlać, gdy na jeden pionowy piksel wykresu przypada więcej niż jedna wartość: **all** - wyświetla najmniejszą, największą i średnią wartość; **min** - wyświetla najmniejszą wartość; **max** - wyświetla największą wartość; **avg** - wyświetla wartość średnią. To ustawienie jest przydatne podczas wyświetlania wykresu dla długiego okresu z częstym interwałem aktualizacji (na przykład wartości z jednego roku zbierane co 10 minut).
Etykieta zestawu danych	Określ etykietę zestawu danych wyświetlaną w konfiguracji wykresu *Zestaw danych* oraz w *Legendzie* wykresu (dla zagregowanych zestawów danych). Wszystkie zestawy danych są numerowane, również te, dla których określono *Etykietę zestawu danych*. Jeśli nie określono etykiety, zestaw danych zostanie oznaczony automatycznie zgodnie z numerem (np. „Zestaw danych nr 2”, „Zestaw danych nr 3” itd.). Numeracja zestawów danych jest przeliczana po zmianie kolejności/przeciągnięciu zestawów danych. Zbyt długie etykiety zestawów danych zostaną skrócone, aby zmieścić się w miejscu wyświetlania (np. „Liczba proc...”).
Tagi pozycji	Określ tagi używane do filtrowania pozycji wyświetlanych w widgecie. Można uwzględniać lub wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnia określone nazwy tagów; **Równa się** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter); **Nie istnieje** - wyklucza określone nazwy tagów; **Nie równa się** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Or; **Or** - wystarczy spełnienie jednego warunku.

Szczegóły konfiguracji zestawu danych

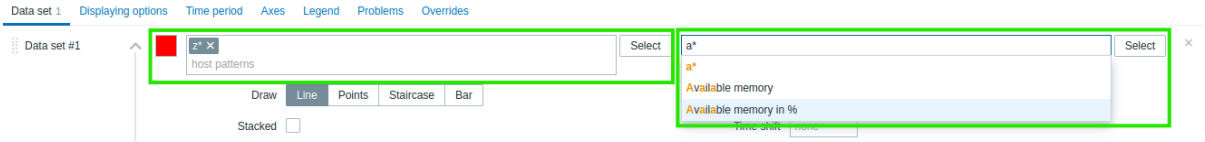
Istniejące zestawy danych są wyświetlane na liście. Możesz:

- Kliknąć ikonę przenoszenia i przeciągnąć zestaw danych w nowe miejsce na liście.
- Kliknąć ikonę rozwijania, aby rozwinąć szczegóły zestawu danych. Po rozwinięciu ta ikona zmienia się w ikonę zwijania.
- Kliknąć ikonę koloru, aby otworzyć selektor. Możesz wprowadzić kod szesnastkowy, wybrać jednolity kolor lub przełączyć się na kartę *Paleta* i wybrać wiersz predefiniowanych kolorów. Wybrany kolor jest stosowany bezpośrednio w zestawach danych typu *Item list* lub jako baza dla generowanych odcieni w zestawach danych typu *Item patterns*. Użyj klawisza Tab, aby przechodzić między elementami sterującymi okna dialogowego, klawiszy strzałek, aby poruszać się po próbkach kolorów lub wierszach palety, klawisza Enter, aby wybrać, oraz klawisza Esc, aby anulować.
- Kliknąć przycisk *Add new data set*, aby dodać pusty zestaw danych umożliwiający wybranie hosta i wzorców pozycji. Jeśli klikniesz ikonę skierowaną w dół obok przycisku *Add new data set*, pojawi się menu rozwijane umożliwiające dodanie nowego zestawu danych typu *Item patterns* lub *Item list*, a także *Clone* bieżącego otwartego zestawu danych. Jeśli wszystkie zestawy danych są zwinięte, opcja *Clone* jest niedostępna.

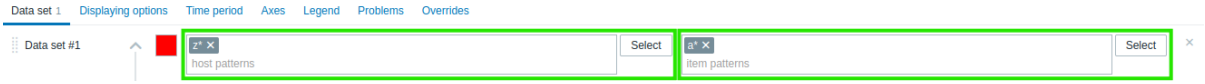


Zestaw danych **Wzorce pozycji** zawiera pola *Wzorce hostów* i *Wzorce pozycji*, które rozpoznają zarówno pełne nazwy, jak i wzorce zawierające symbol wieloznaczny (*). Ta funkcjonalność pozwala wybrać wszystkie nazwy hostów i nazwy pozycji zawierające wybrany wzorzec. Podczas wpisywania nazwy pozycji lub wzorca pozycji w polu *Wzorce pozycji* na liście rozwijanej wyświetlane są tylko pozycje należące do wybranych nazw hostów.

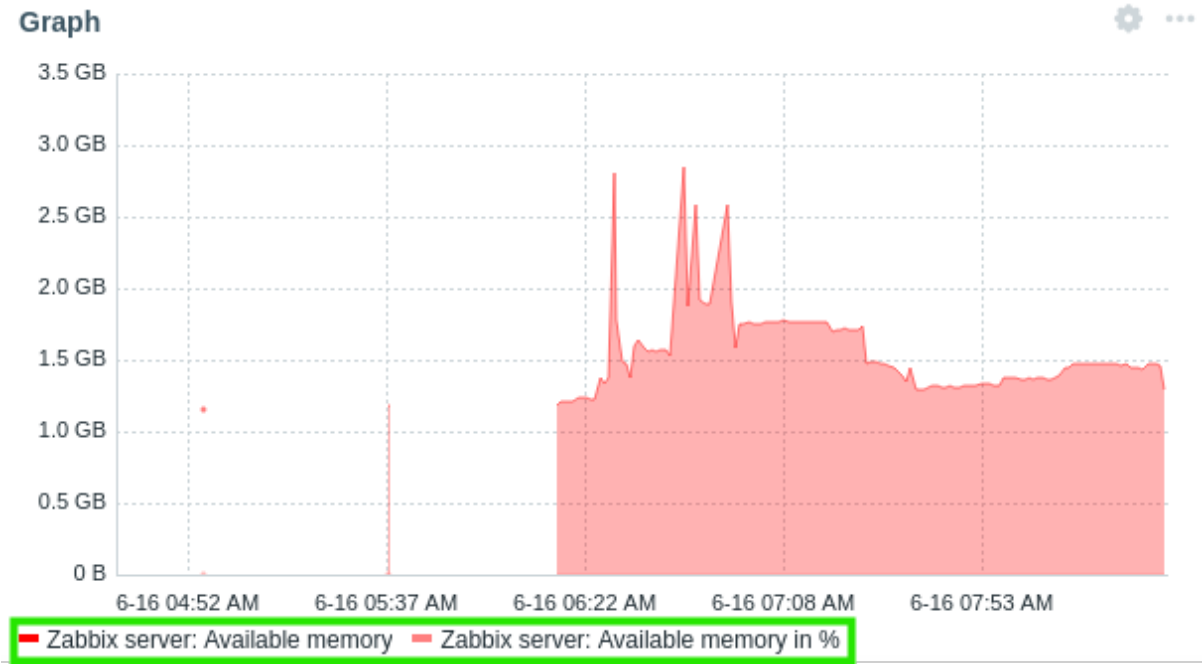
Na przykład po wpisaniu wzorca **z*** w polu *Wzorce hostów* lista rozwijana wyświetla wszystkie nazwy hostów zawierające ten wzorzec: **z*, Zabbix server, Zabbix proxy**. Po naciśnięciu *Enter* wzorzec zostaje zaakceptowany i jest wyświetlany jako **z***. Podobnie po wpisaniu wzorca **a*** w polu *Wzorce pozycji* lista rozwijana wyświetla wszystkie nazwy pozycji zawierające ten wzorzec: **a*, Available memory, Available memory in %**.



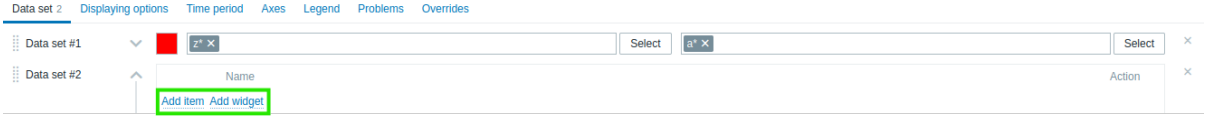
Po naciśnięciu *Enter* wzorzec zostaje zaakceptowany i wyświetlony jako **a***.



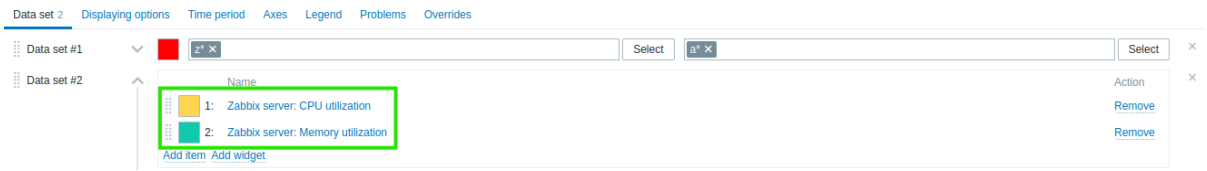
Wykres wyświetla następnie wszystkie pozycje należące do wybranych nazw hostów.



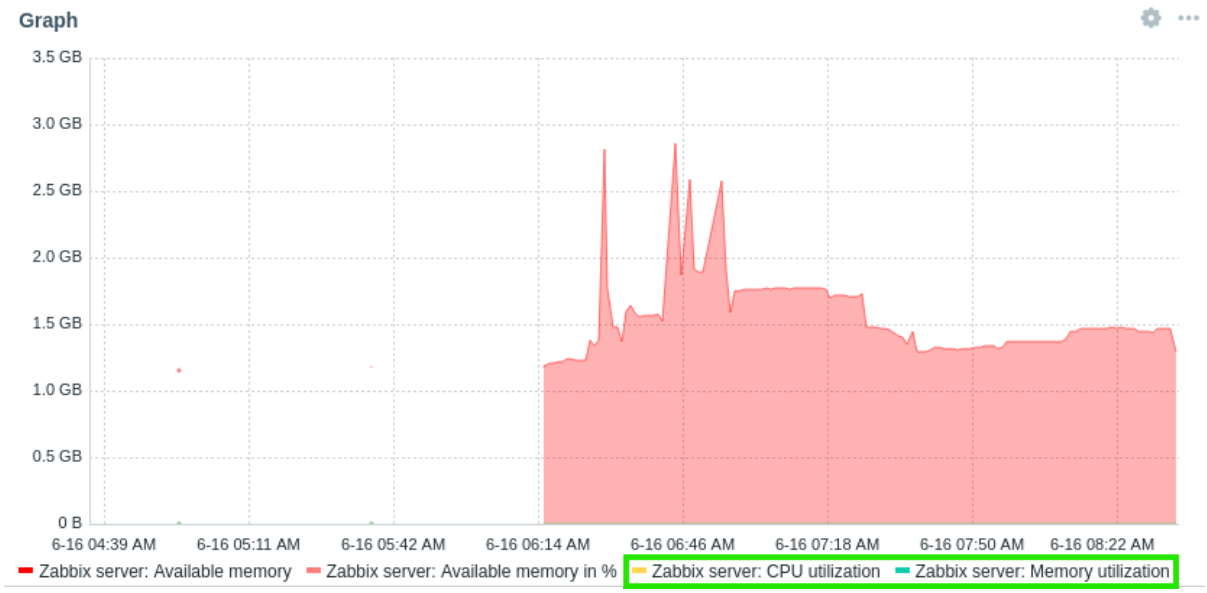
Zestaw danych **Lista pozycji** zawiera przycisk *Dodaj pozycję*, który umożliwia dodanie pozycji wyświetlanych na wykresie. Możesz także dodać zgodne widgety jako źródło danych dla pozycji, klikając przycisk *Dodaj widget*.



Na przykład kliknięcie przycisku *Dodaj pozycję* otwiera wyskakujące okno zawierające parametr *Host*. Po wybraniu hosta na liście wyświetlane są wszystkie jego pozycje dostępne do wyboru.



Po wybraniu jednej lub większej liczby pozycji zostaną one wyświetlone na liście pozycji zestawu danych oraz na wykresie.

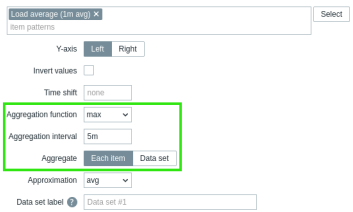


Agregacja na wykresach

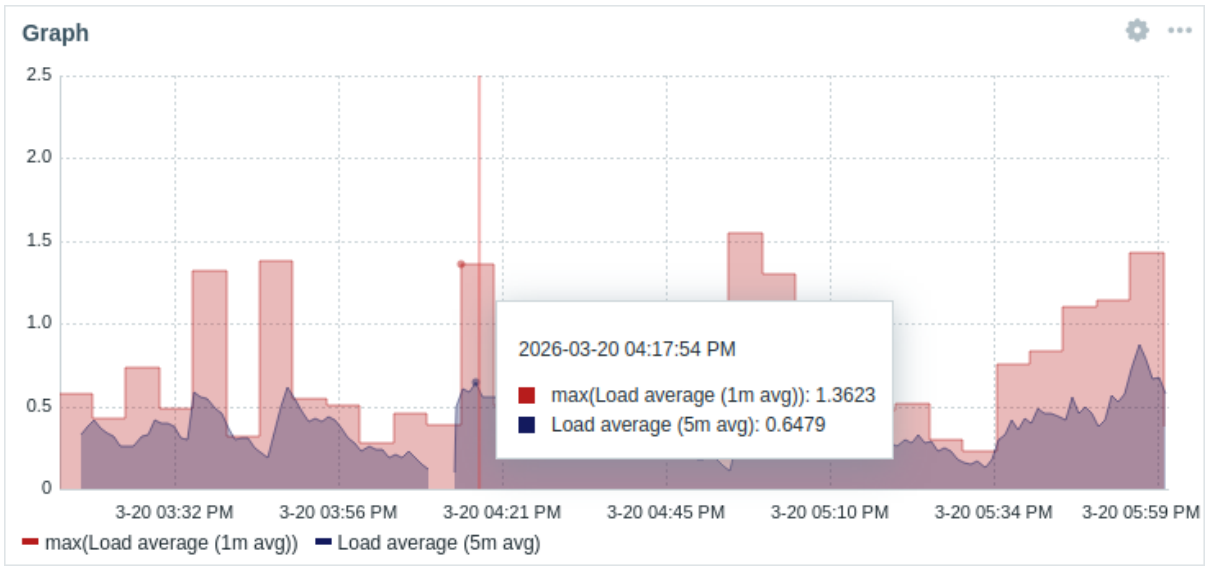
Agregacja umożliwia wyświetlenie zagregowanej wartości (minimalnej, maksymalnej, średniej itd.) dla określonego przedziału czasu (5 minut, godziny, dnia itd.) zamiast wszystkich pojedynczych wartości. W obrębie zestawu danych agregację można zastosować do każdej pozycji osobno lub do całego zestawu danych.

Przykładowe zastosowania funkcji agregacji: - Średnia liczba żądań do serwera NGINX dziennie. - Najmniejsza ilość wolnego miejsca na dysku wśród klastrów w tygodniu.

W poniższym przykładzie do pozycji zastosowano funkcję agregacji „max” z pięciominutowym przedziałem agregacji:



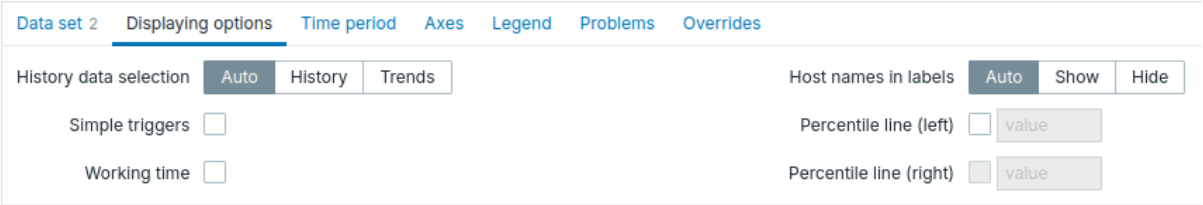
Przykład przedstawia maksymalne obciążenie procesora w ciągu pięciu minut (zagregowane na podstawie jednominutowych wartości średnich) w porównaniu ze średnim obciążeniem procesora w ciągu pięciu minut.



Po wybraniu opcji *Pokaż funkcję agregacji* dla legendy pozycje są wyświetlane w nawiasach, a w legendzie i etykiecie narzędzia są poprzedzone funkcją agregacji. Jeśli dla wykresu określono *Etykietę zestawu danych*, etykieta jest wyświetlana w nawiasach i poprzedzona funkcją agregacji.

Opcje wyświetlania

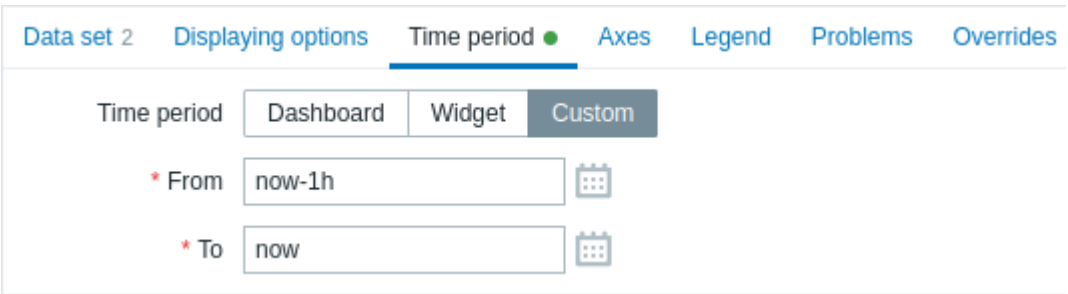
Karta **Opcje wyświetlania** umożliwia zdefiniowanie wyboru danych historycznych:



Wybór danych historycznych	Ustaw źródło danych wykresu: **Auto** - dane są pobierane zgodnie z klasycznym algorytmem wykresu (domyślnie); **Historia** - dane z historii; **Trendy** - dane z trendów.
Proste wyzwalacze	Zaznacz pole wyboru, aby wyświetlić progi wyzwalaczy dla prostych wyzwalaczy. Progi zostaną narysowane jako linie przerywane w kolorze ważności wyzwalacza. Prosty wyzwalacz to wyzwalacz z jedną funkcją (tylko `last`, `max`, `min`, `avg`) dla jednej pozycji w wyrażeniu. Można narysować maksymalnie trzy wyzwalacze. Pamiętaj, że wyzwalacz musi znajdować się w obrębie rysowanego zakresu, aby był widoczny.
Czas pracy	Zaznacz pole wyboru, aby wyświetlić czas pracy na wykresie. Czas pracy (dni robocze) jest wyświetlany na wykresach jako białe tło, natomiast czas wolny jest wyświetlany na szaro (w przypadku domyślnego motywu frontendu *Original blue*).
Nazwy hostów na etykietach	Wybierz, czy wyświetlać nazwy hostów w legendzie wykresu i w etykiecie narzędzia wyświetlanej po umieszczeniu wskaźnika nad danymi wykresu: **Auto** - nazwy hostów są wyświetlane tylko wtedy, gdy w zestawach danych znajduje się więcej niż jeden host (domyślnie). **Pokaż** - nazwy hostów są wyświetlane. **Ukryj** - nazwy hostów są ukryte. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu.
Linia percentyla (lewa)	Zaznacz pole wyboru i wprowadź wartość percentyla, aby wyświetlić określony percentyl jako linię na lewej osi Y wykresu. Jeśli na przykład ustawiono percentyl 95%, linia percentyla będzie znajdować się na poziomie, poniżej którego mieści się 95 procent wartości.
Linia percentyla (prawa)	Zaznacz pole wyboru i wprowadź wartość percentyla, aby wyświetlić określony percentyl jako linię na prawej osi Y wykresu. Jeśli na przykład ustawiono percentyl 95%, linia percentyla będzie znajdować się na poziomie, poniżej którego mieści się 95 procent wartości.

Okres

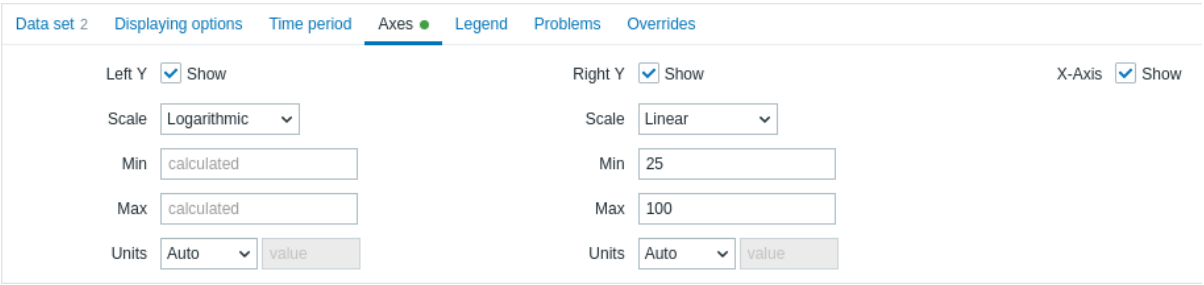
Karta **Okres** umożliwia ustawienie okresu, dla którego mają być wyświetlane dane na wykresie:



Okres	Wybierz źródło danych dla okresu: **Dashboard** - użyj selektora okresu dashboardu; **Widget** - użyj kompatybilnego widgetu (ustawionego w parametrze *Widget*); **Niestandardowy** - użyj niestandardowego okresu ustawionego w parametrach *Od* i *Do*; jeśli jest ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara wskazująca ustawiony czas po najechaniu kursorem myszy. Należy pamiętać, że niezależnie od konfiguracji parametru *Okres* widgetu kompatybilne widgety nadal mogą używać go jako źródła danych dla okresu.
Widget	Wprowadź lub wybierz kompatybilny widget jako źródło danych dla okresu. Ten parametr jest dostępny, jeśli dla parametru *Okres* ustawiono wartość „Widget”.
Od	Wprowadź lub wybierz początek okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla parametru *Okres* ustawiono wartość „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla parametru *Okres* ustawiono wartość „Niestandardowy”.

Osie

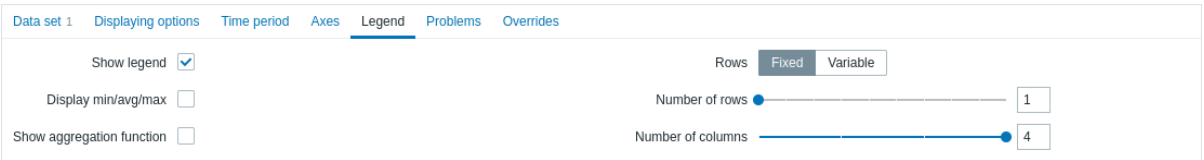
Karta **Osie** umożliwia dostosowanie sposobu wyświetlania osi:



Lewa oś Y	Zaznacz to pole wyboru, aby wyświetlić lewą oś Y. Pole wyboru może być nieaktywne, jeśli nie zostało zaznaczone na karcie *Zestaw danych* lub *Zastąpienia*.
Prawa oś Y	Zaznacz to pole wyboru, aby wyświetlić prawą oś Y. Pole wyboru może być nieaktywne, jeśli nie zostało zaznaczone na karcie *Zestaw danych* lub *Zastąpienia*.
Oś X	Odznacz to pole wyboru, aby ukryć oś X (domyślnie zaznaczone).
Skala	Wybierz skalę wartości osi wykresu z listy rozwijanej: **Liniowa** - wartości osi zwiększają się o stałą wartość (np. 10, 20, 30); skala jest odpowiednia dla danych zmieniających się równomiernie lub obejmujących mały albo umiarkowany zakres; **Logarytmiczna** - wartości osi zwiększają się wykładniczo (np. 10, 100, 1000); skala jest odpowiednia dla danych zmieniających się szybko lub obejmujących duży zakres.
Min	Ustaw minimalną wartość odpowiedniej osi. Określana jest minimalna wartość widocznego zakresu osi Y.
Maks.	Ustaw maksymalną wartość odpowiedniej osi. Określana jest maksymalna wartość widocznego zakresu osi Y.
Jednostki	Wybierz jednostkę wartości osi wykresu z listy rozwijanej: **Automatycznie** - wartości osi są wyświetlane przy użyciu jednostki pierwszej pozycji w zestawie danych; **Statyczne** - wartości osi są wyświetlane przy użyciu jednostki określonej w polu wprowadzania *wartość*; jeśli pole pozostanie puste, wyświetlane będą tylko wartości liczbowe.

Legenda

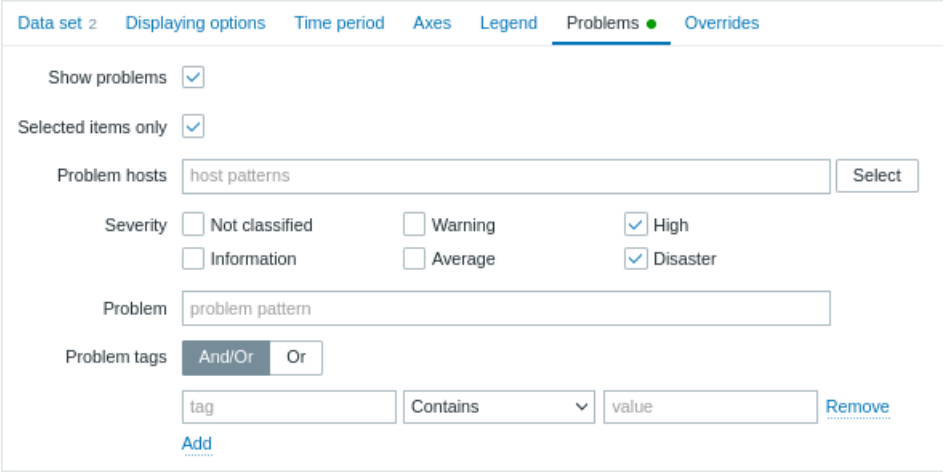
Karta **Legenda** umożliwia dostosowanie legendy wykresu:



Pokaż legendę	Odznacz to pole wyboru, aby ukryć legendę na wykresie (domyślnie zaznaczone).
Wyświetl min./śr./maks.	Zaznacz to pole wyboru, aby wyświetlić w legendzie minimalne, średnie i maksymalne wartości pozycji.
Pokaż funkcję agregacji	Zaznacz to pole wyboru, aby wyświetlić funkcję agregacji w legendzie.
Wiersze	Wybierz tryb wyświetlania wierszy legendy: **Stała** - liczba wyświetlanych wierszy jest określana przez wartość parametru *Liczba wierszy*; **Zmienna** - liczba wyświetlanych wierszy jest określana przez liczbę skonfigurowanych pozycji, ale nie może przekraczać wartości parametru *Maksymalna liczba wierszy*.
Liczba wierszy/ Maksymalna liczba wierszy	Jeśli dla opcji *Wiersze* ustawiono wartość „Stała”, ustaw liczbę wyświetlanych wierszy legendy (1-10). Jeśli dla opcji *Wiersze* ustawiono wartość „Zmienna”, ustaw maksymalną liczbę wyświetlanych wierszy legendy (1-10).
Liczba kolumn	Ustaw liczbę wyświetlanych kolumn legendy (1-4). Ten parametr jest dostępny, jeśli opcja *Wyświetl min./śr./maks.* nie jest zaznaczona.

Problemy

Karta **Problemy** umożliwia dostosowanie sposobu wyświetlania problemów:

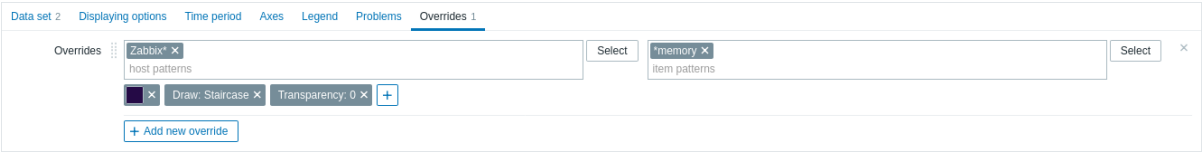


Pokaż problemy	Zaznacz to pole wyboru, aby włączyć wyświetlanie problemów na wykresie (domyślnie niezaznaczone, czyli wyłączone).
Tylko wybrane pozycje	Zaznacz to pole wyboru, aby na wykresie wyświetlane były tylko problemy dotyczące wybranych pozycji.
Hosty z problemami	Wybierz hosty, na których występują problemy, które mają być wyświetlane na wykresie. Można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub większej liczby znaków). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij klawisz *Enter*. Podczas wpisywania zwróć uwagę, że wszystkie pasujące hosty są wyświetlane na liście rozwijanej. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Ważność	Zaznacz poziomy ważności problemów, aby filtrować problemy wyświetlane na wykresie. Jeśli nie zaznaczono żadnego poziomu ważności, wyświetlane będą wszystkie problemy.
Problem	Określ nazwę problemu, który ma być wyświetlany na wykresie.

Tagi problemu	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Można uwzględniać i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnia określone nazwy tagów; **Równa się** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter); **Nie istnieje** - wyklucza określone nazwy tagów; **Nie równa się** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem *Or*; **Or** - wystarczy spełnienie jednego warunku.

Nadpisywanie

Karta **Nadpisywanie** umożliwia dodawanie niestandardowych nadpisań dla zestawów danych:



Nadpisania są przydatne, gdy dla zestawu danych wybrano kilka pozycji za pomocą symbolu wieloznacznego ∗ i chcesz zmienić sposób ich domyślnego wyświetlania (np. domyślny kolor bazowy lub dowolną inną właściwość).

Istniejące nadpisania (jeśli są dostępne) są wyświetlane na liście. Aby dodać nowe nadpisanie:

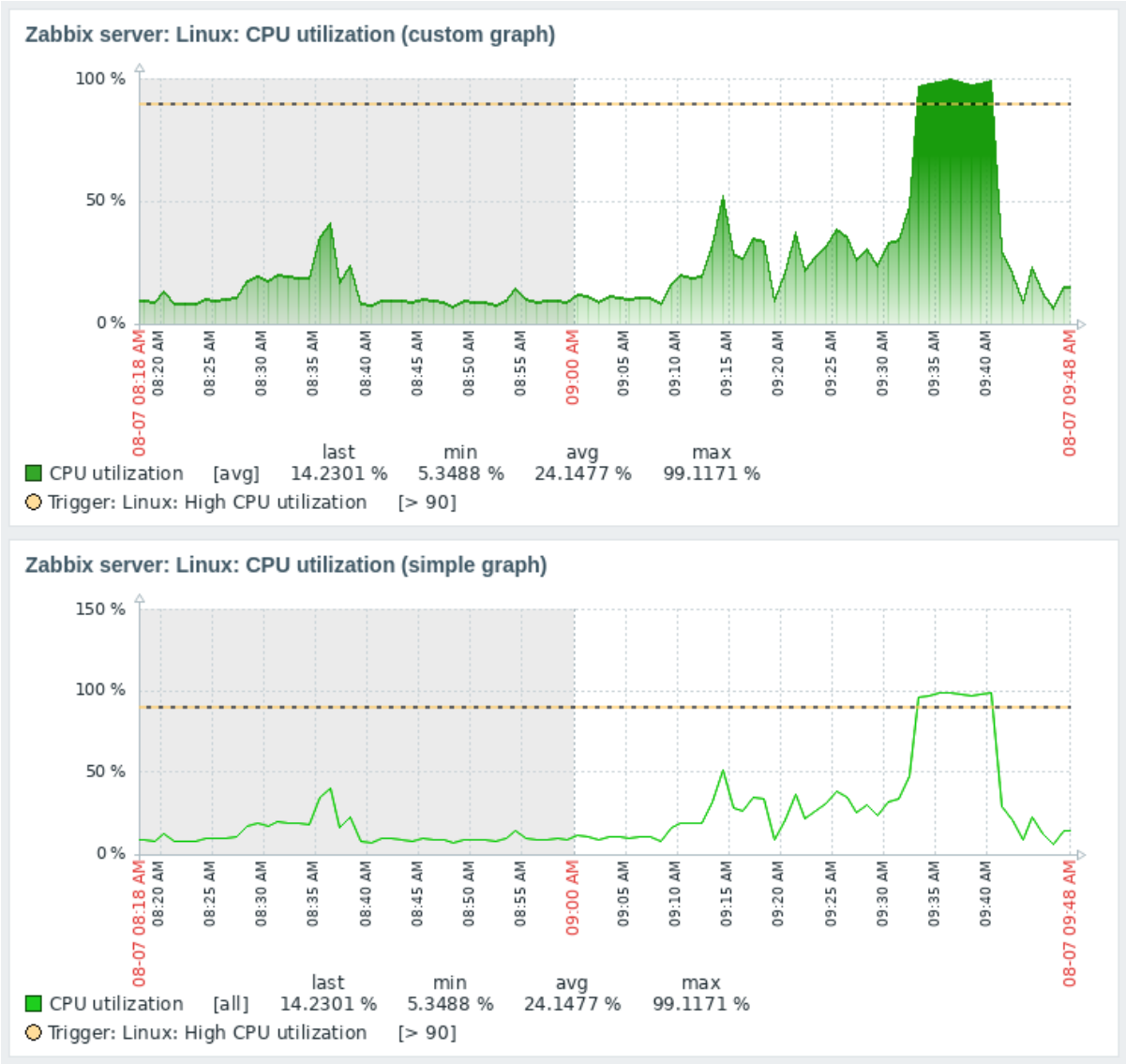
- Kliknij przycisk 

- Wybierz hosty i pozycje dla nadpisania. Alternatywnie możesz wprowadzić wzorce hostów i pozycji. Można używać wzorców z symbolami wieloznacznymi (na przykład ∗ zwróci wyniki pasujące do zera lub większej liczby znaków). Aby określić wzorzec z symbolem wieloznacznym, wprowadź ciąg ręcznie i naciśnij *Enter*. Podczas wpisywania zwróć uwagę, że wszystkie pasujące hosty są wyświetlane na liście rozwijanej. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać indywidualnie na przykład pozycji o nazwie ”item*”, jeśli istnieją inne pasujące pozycje (np. item2, item3). Parametry wzorca hosta i wzorca pozycji są wymagane. Parametr służący do określania wzorców hostów nie jest dostępny podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu. Parametr służący do określania listy pozycji umożliwia wyłącznie wybór pozycji skonfigurowanych w szablonie podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu.
- Kliknij  , aby wybrać parametry nadpisania. Należy wybrać co najmniej jeden parametr nadpisania. Opisy parametrów znajdziesz powyżej, na karcie *Zestaw danych*.

Wykres (klasyczny)

Przegląd

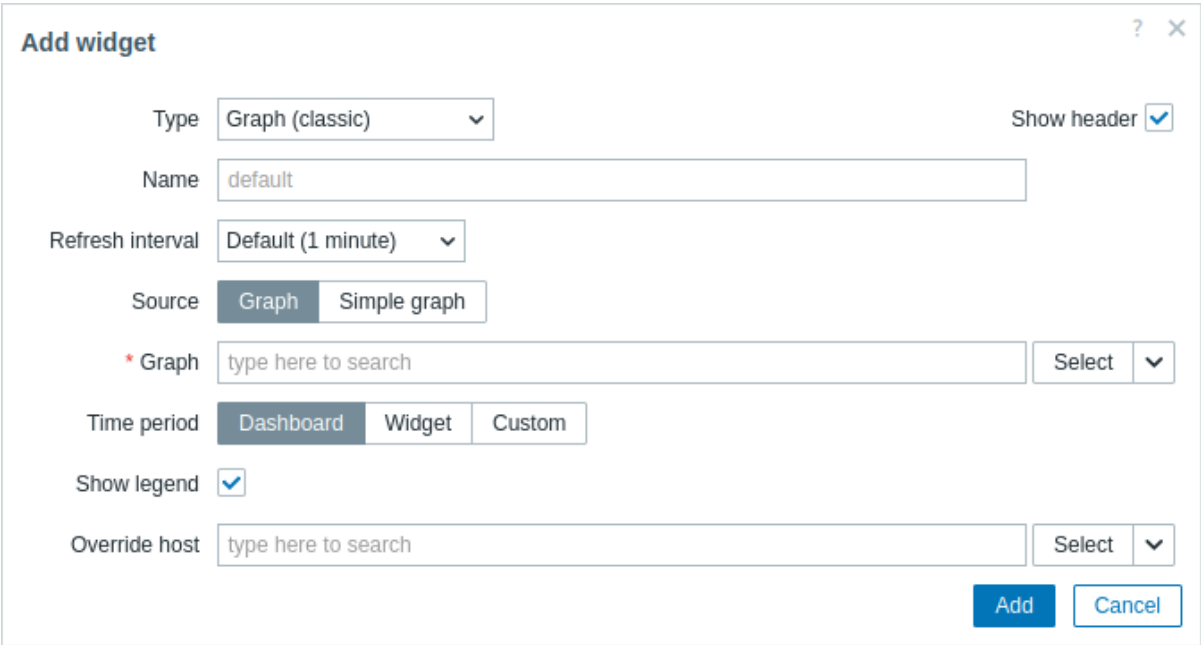
Widżet *Wykres (klasyczny)* wyświetla dane liczbowe z pozycji jako obrazowy wykres niestandardowy lub wykres prosty. Obsługuje wybór okresu, linie wyzwalaczy oraz opcjonalne wyświetlanie legendy.



Informacje wyświetlane w widżecie *Wykres (klasyczny)* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu *Działania* widżetu ⋯, które pojawia się po najechaniu kursorem na widżet. W zależności od ustawień przeglądarki obraz jest automatycznie zapisywany w folderze `Downloads` albo otwierane jest okno dialogowe *Zapisz*, umożliwiające wybranie lokalizacji.

Konfiguracja

Aby skonfigurować widget, wybierz *Graph (classic)* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Source	Wybierz typ wykresu: **Graph** - wykres niestandardowy; **Simple graph** - wykres prosty.
Graph	Wybierz wykres niestandardowy do wyświetlenia. Alternatywnie wybierz kompatybilny widget jako źródło danych dla wykresów. Ten parametr jest dostępny, jeśli *Source* jest ustawione na ”Graph”.
Item	Wybierz pozycja do wyświetlenia na wykresie prostym. Alternatywnie wybierz kompatybilny widget jako źródło danych dla pozycji. Ten parametr jest dostępny, jeśli *Source* jest ustawione na ”Simple graph”.
Time period	Ustaw przedział czasu, dla którego mają być wyświetlane dane na wykresie. Wybierz źródło danych dla przedziału czasu: **Dashboard** - użyj selektora przedziału czasu pulpitu; **Widget** - użyj kompatybilnego widgetu (ustawionego w parametrze *Widget*); **Custom** - użyj niestandardowego przedziału czasu ustawionego w parametrach *From* i *To*; jeśli jest ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem. Zwróć uwagę, że niezależnie od konfiguracji *Time period* widgetu, kompatybilne widgety nadal mogą używać go jako źródła danych dla przedziału czasu.
Widget	Wprowadź lub wybierz kompatybilny widget jako źródło danych dla przedziału czasu. Ten parametr jest dostępny, jeśli *Time period* jest ustawione na ”Widget”.
From	Wprowadź lub wybierz początek przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawione na ”Custom”.
To	Wprowadź lub wybierz koniec przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawione na ”Custom”.
Show legend	Odznacz to pole wyboru, aby ukryć legendę na wykresie (domyślnie zaznaczone).
Override host	Wybierz kompatybilny widget lub selektor hostów pulpitu jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.

Legenda wykresu składa się z trzech sekcji:

- Pozycje i ich wartości zagregowane
- Percentyle (jeśli są skonfigurowane)
- Wyzwalacze (jeśli są powiązane z wyświetlanymi pozycjami)

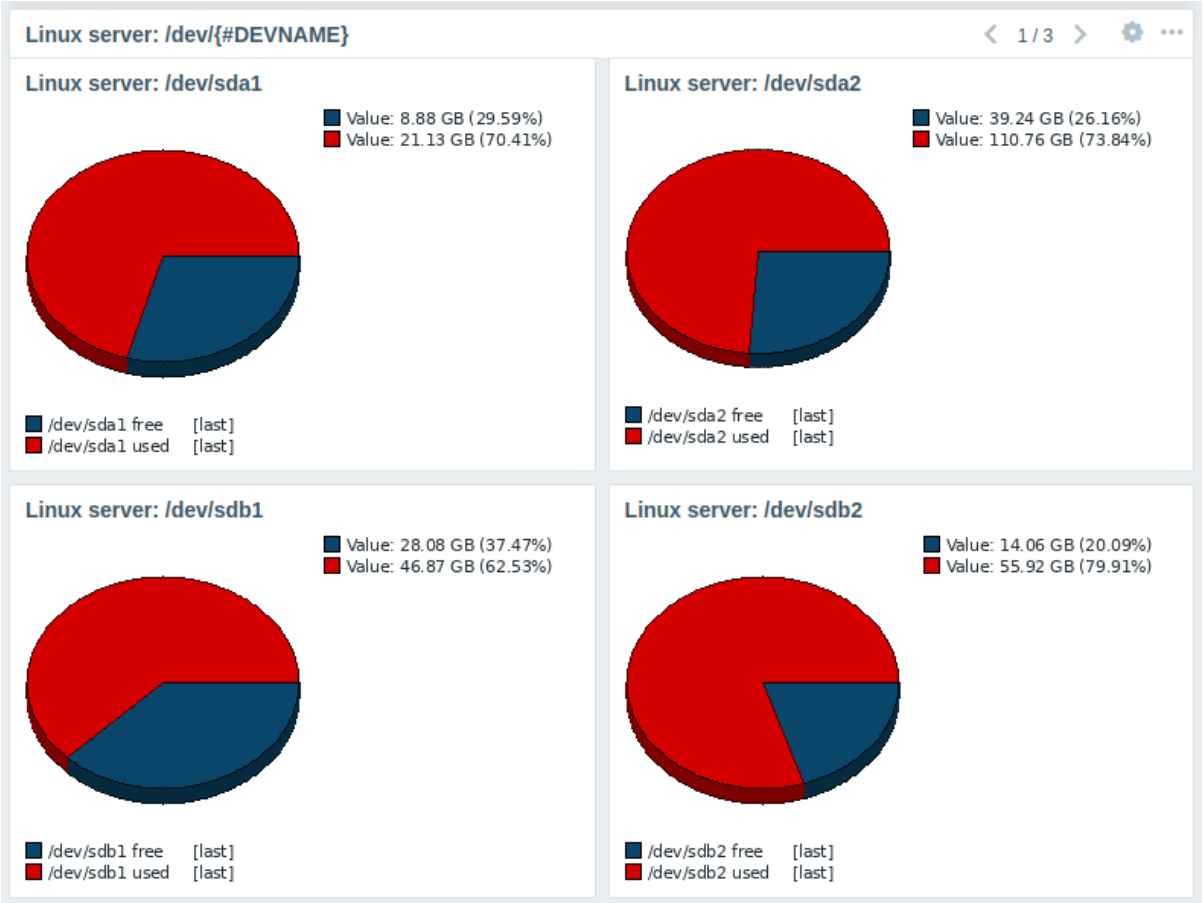
Jeśli wysokość wykresu w obrębie widgetu jest niewystarczająca, legenda może nie być wyświetlana lub może być wyświetlana tylko częściowo. Najpierw ukrywane są wyzwalacze i percentyle, a następnie legenda pozycji. Aby wyświetlić pełną legendę, zwiększ pionowy rozmiar widgetu.

Note:
Można wyświetlić nie więcej niż 3 linie wyzwalaczy. Jeśli wyzwalaczy jest więcej, priorytet wyświetlania mają wyzwalacze o niższym poziomie ważności.

Prototyp wykresu

Przegląd

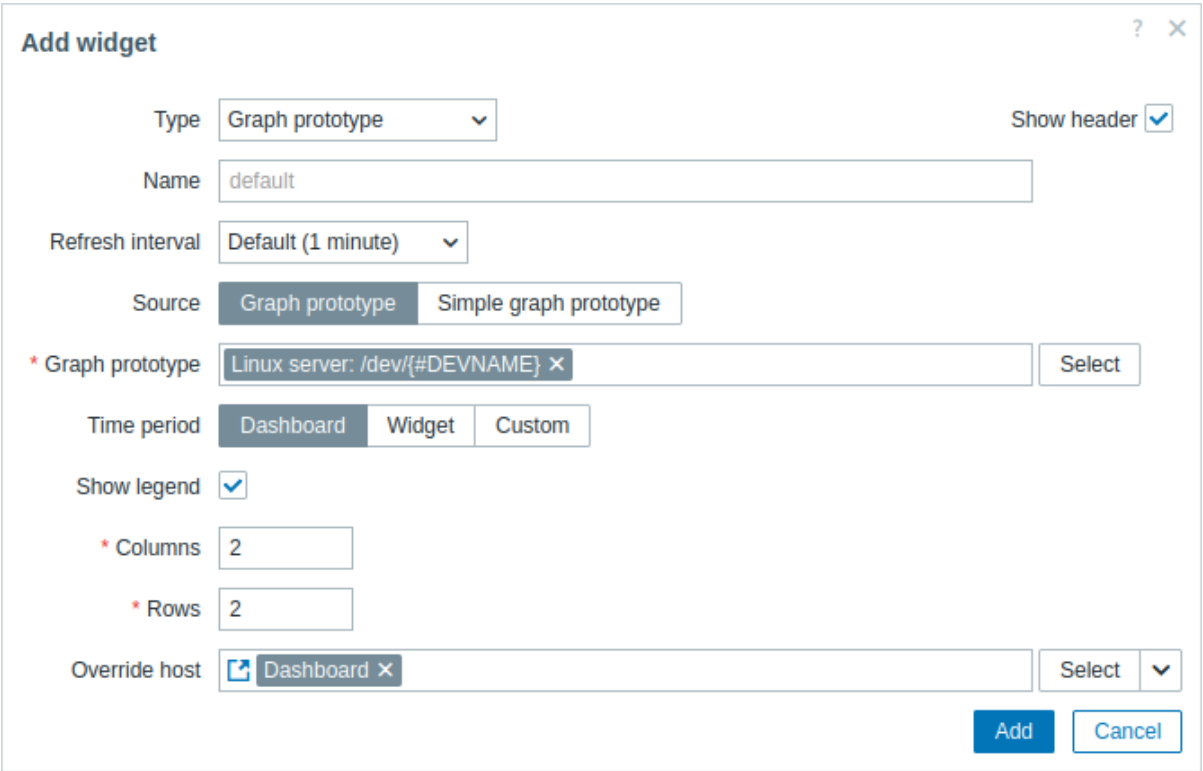
Widżet *Prototyp wykresu* wyświetla siatkę automatycznie tworzonych wykresów niestandardowych na podstawie prototypów wykresów lub prototypów pozycji reguły wykrywania niskiego poziomu.



Wykresy są rozmieszczane w konfigurowalnym układzie siatki; jeśli liczba wykresów przekracza pojemność układu, w widżecie włączane jest stronicowanie.

Konfiguracja

Aby skonfigurować widget, wybierz *Prototyp wykresu* jako typ widgetu:



Oprócz parametrów wspólnych dla wszystkich widgetów możesz ustawić następujące opcje:

Źródło	Wybierz źródło wykresów: **Prototyp wykresu** lub **Prosty prototyp wykresu**.

Prototyp wykresu	Wybierz prototyp wykresu, aby wyświetlać wykresy wykryte przez prototyp wykresu. Ten parametr jest dostępny, jeśli dla opcji *Źródło* ustawiono wartość „Prototyp wykresu”.
Prototyp pozycji	Wybierz prototyp pozycji, aby wyświetlać proste wykresy dla pozycji wykrytych przez prototyp pozycji. Ten parametr jest dostępny, jeśli dla opcji *Źródło* ustawiono wartość „Prosty prototyp wykresu”.
Okres	Ustaw okres, dla którego mają być wyświetlane dane na wykresach. Wybierz źródło danych dla okresu: **Dashboard** — użyj selektora okresu dashboardu; **Widget** — użyj zgodnego widgetu (ustawionego w parametrze *Widget*); **Niestandardowy** — użyj niestandardowego okresu ustawionego w parametrach *Od* i *Do*; jeśli zostanie ustawiony, w prawym górnym rogu widgetu będzie wyświetlana ikona zegara wskazująca ustawiony czas po najechaniu kursorem. Należy pamiętać, że niezależnie od konfiguracji opcji *Okres* widgetu zgodne widgety nadal mogą używać go jako źródła danych dla okresu.
Widget	Wprowadź lub wybierz zgodny widget jako źródło danych dla okresu. Ten parametr jest dostępny, jeśli dla opcji *Okres* ustawiono wartość „Widget”.
Od	Wprowadź lub wybierz początek okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla opcji *Okres* ustawiono wartość „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla opcji *Okres* ustawiono wartość „Niestandardowy”.
Pokaż legendę	Odznacz to pole wyboru, aby ukryć legendę na wykresach (domyślnie zaznaczone).
Zastąp host	Wybierz zgodny widget lub selektor hostów dashboardu jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na dashboardzie szablonu.
Kolumny	Wprowadź liczbę kolumn wykresów, które mają być wyświetlane w widgecie prototypu wykresu.
Wiersze	Wprowadź liczbę wierszy wykresów, które mają być wyświetlane w widgecie prototypu wykresu.

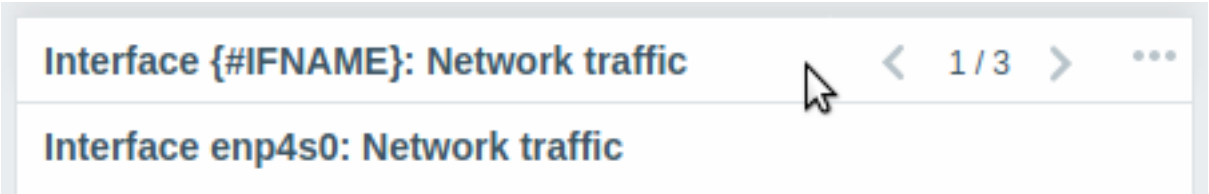
Legenda wykresu składa się z trzech sekcji:

- Pozycje i ich zagregowane wartości
- Percentyle (jeśli skonfigurowano)
- Wyzwalacze (jeśli są powiązane z wyświetlanymi pozycjami)

Jeśli wysokość wykresu w widgecie jest niewystarczająca, legenda może nie być wyświetlana lub może być wyświetlana tylko częściowo. Najpierw ukrywane są wyzwalacze i percentyle, a następnie legenda pozycji. Aby wyświetlić pełną legendę, zwiększ wysokość widgetu.

Note:
Można wyświetlić nie więcej niż 3 wiersze wyzwalaczy. Jeśli wyzwalaczy jest więcej, do wyświetlenia w pierwszej kolejności wybierane są wyzwalacze o niższym poziomie ważności.

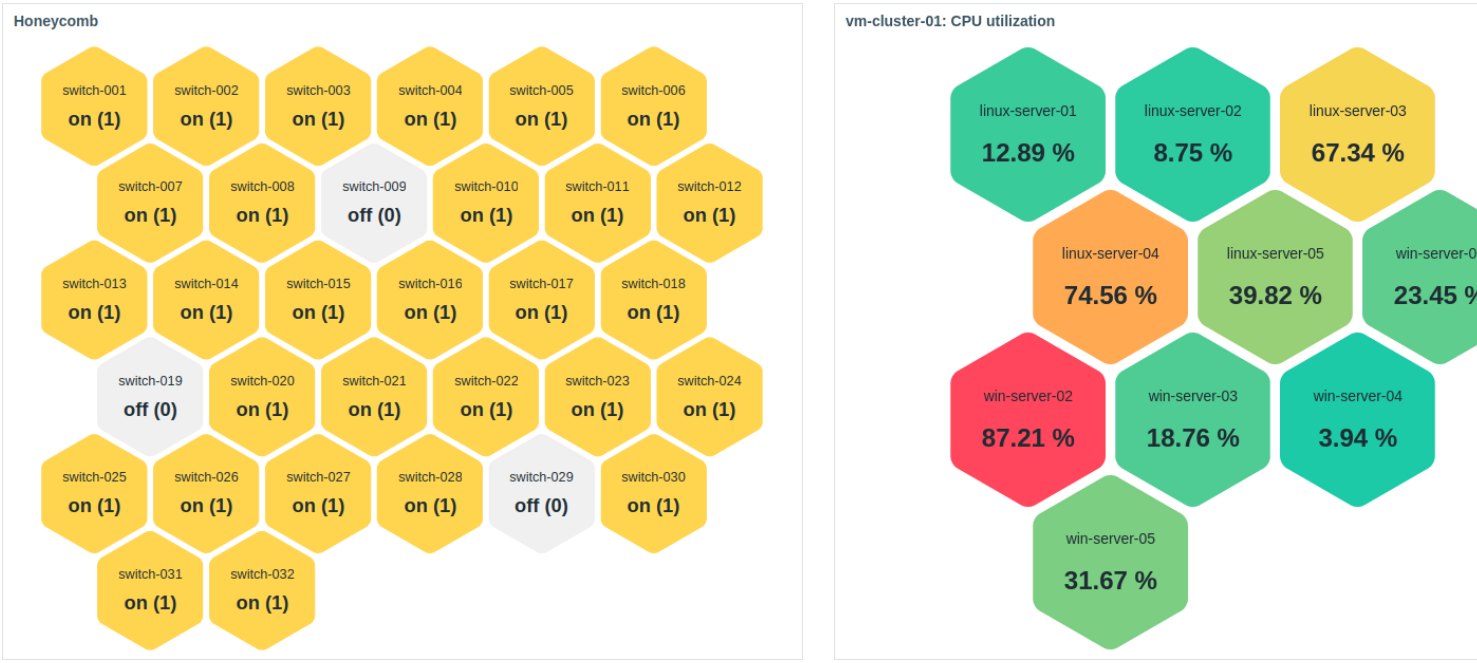
Chociaż parametry *Kolumny* i *Wiersze* umożliwiają dopasowanie więcej niż jednego wykresu do widgetu, liczba wykrytych wykresów może nadal przekraczać liczbę kolumn lub wierszy widgetu. W takim przypadku w widgecie staje się dostępna paginacja, a wysuwany nagłówek umożliwia przełączanie między stronami za pomocą strzałek w lewo i w prawo:



Honeycomb

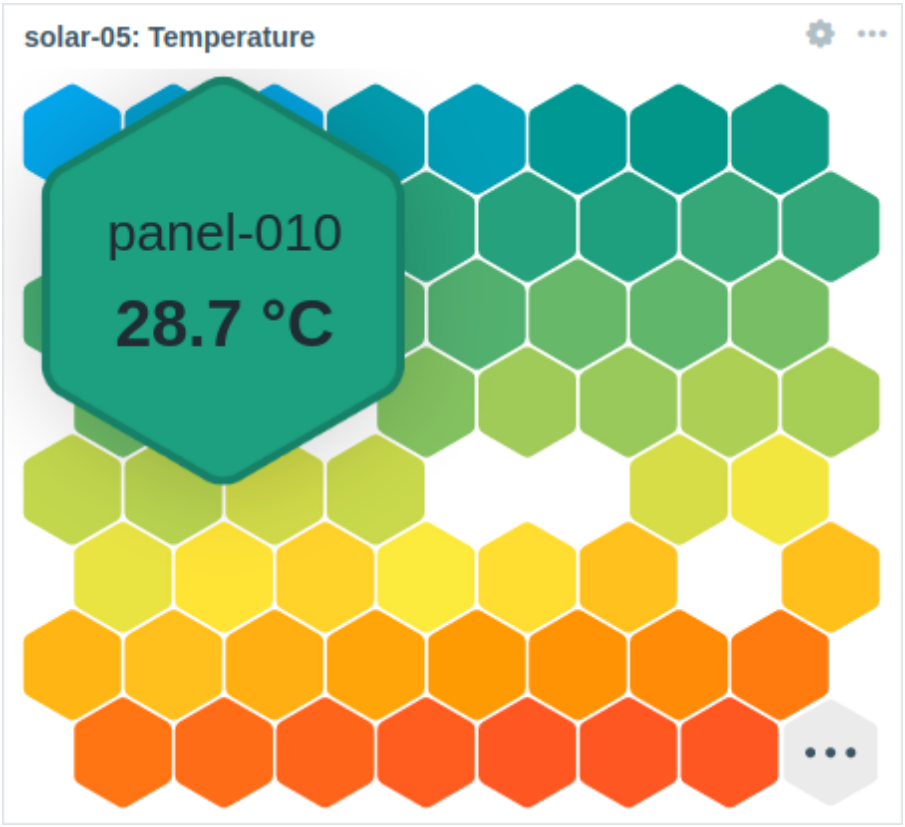
Przegląd

Widżet *Honeycomb* wyświetla hosty lub pozycje jako sześciokątne komórki w siatce. Taki układ ułatwia wykrywanie problematycznych hostów/pozycji, dostrzeganie skupisk problemów, porównywanie grup na pierwszy rzut oka i nie tylko.



Liczba wyświetlanych komórek jest ograniczona rozmiarem widżetu i minimalnym rozmiarem komórki (32px). Jeśli nie wszystkie komórki się mieszczą, ostatnia komórka wyświetla wielokropek. Widżet można zmieniać rozmiar, a liczba, rozmiar i położenie komórek są dostosowywane dynamicznie.

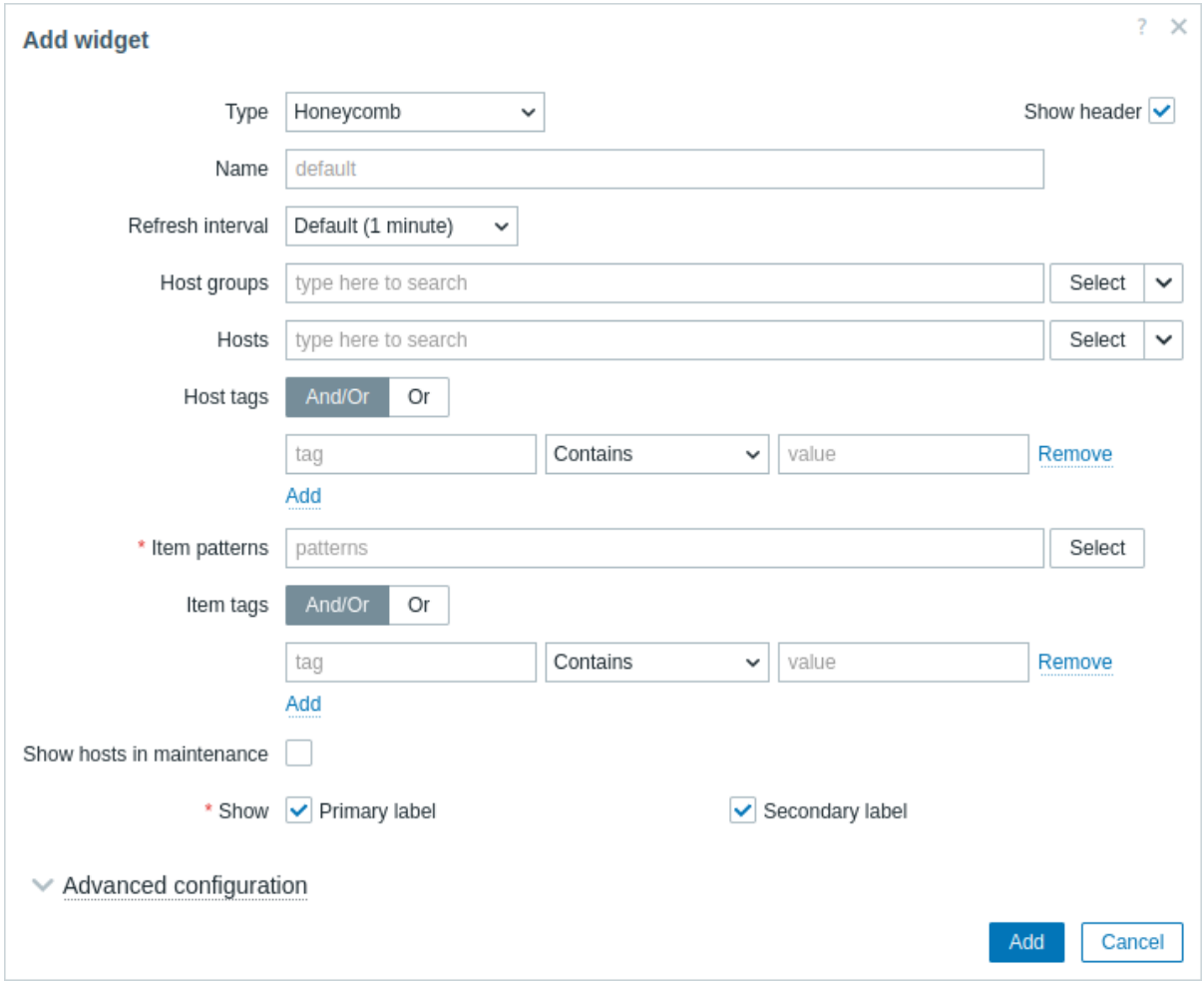
Po najechaniu kursorem myszy aktywna komórka powiększa się, co poprawia jej widoczność. Kliknięcie komórki wyróżnia jej obramowanie do momentu wybrania innej komórki.



Informacje wyświetlane w widżecie *Honeycomb* można pobrać jako obraz PNG, wybierając opcję *Download image* w menu *Actions* widżetu ..., które pojawia się po najechaniu kursorem na widżet. W zależności od ustawień przeglądarki obraz jest automatycznie zapisywany w folderze `Downloads` albo otwiera się okno dialogowe *Save*, umożliwiające wybranie lokalizacji.

Konfiguracja

Aby skonfigurować, wybierz *Honeycomb* jako typ:



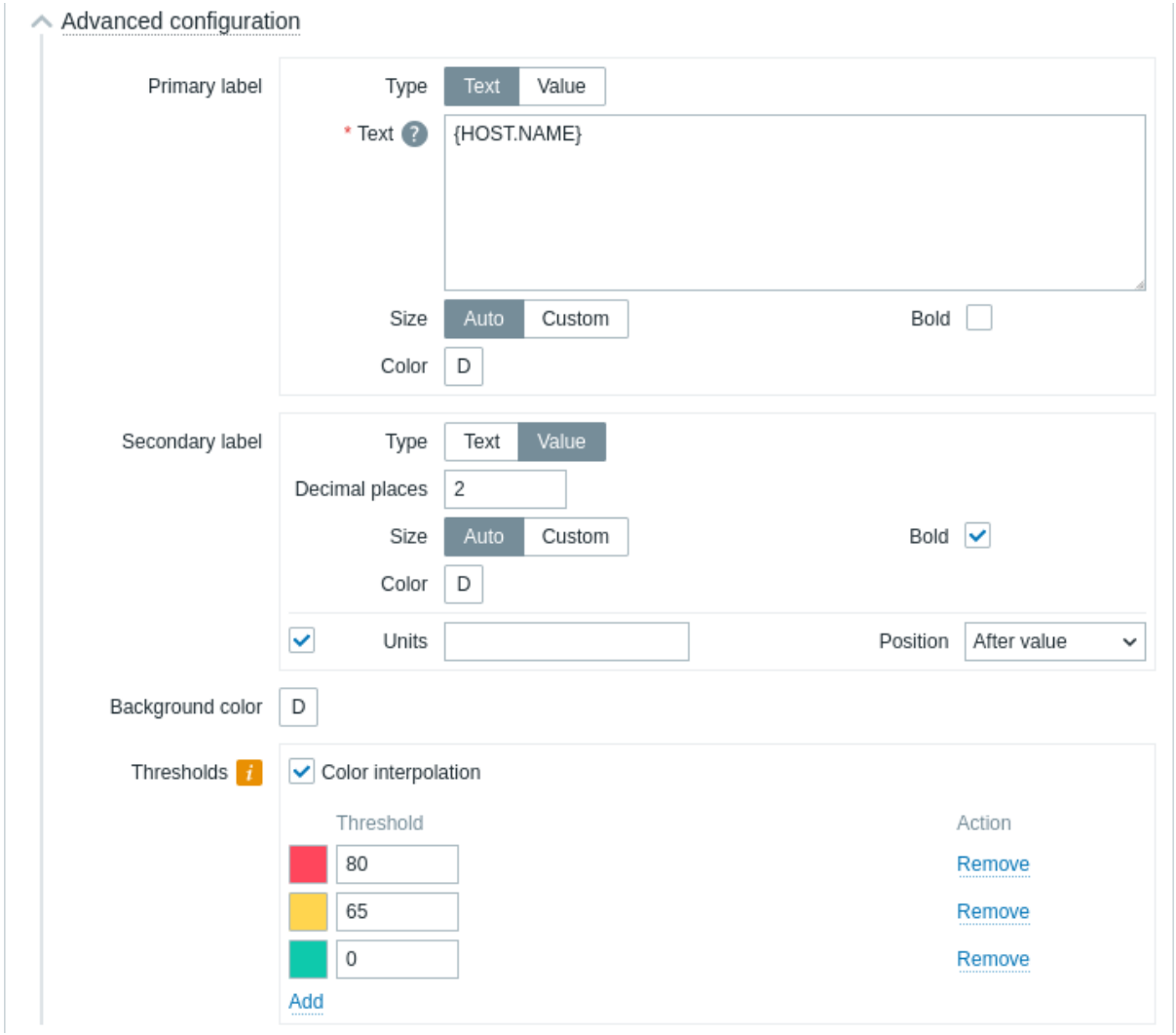
Oprócz parametrów wspólnych dla wszystkich widżetów można ustawić następujące opcje specyficzne:

Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Wybranie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów; jeśli nie wybrano żadnych grup hostów, widżet wyświetli wszystkie grupy hostów zawierające hosty z pozycjami pasującymi do wybranego *Wzorca pozycji* (zobacz poniżej). Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	Wybierz hosty. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli nie wybrano żadnych hostów, widżet wyświetli wszystkie hosty z pozycjami pasującymi do wybranego *Wzorca pozycji* (zobacz poniżej). Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

Tagi hostów	Określ tagi, aby ograniczyć liczbę hostów wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** - wystarczy spełnienie jednego warunku. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Wzorce pozycji	Wprowadź wzorce pozycji lub wybierz istniejące pozycje jako wzorce pozycji. Dane pozycji pasujących do wprowadzonych lub wybranych wzorców będą wyświetlane w widoku honeycomb. Parametr *Wzorce pozycji* jest obowiązkowy. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci pozycje pasujące do zera lub większej liczby znaków; `Zabbix*` zwróci pozycje rozpoczynające się od „Zabbix”). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij *Enter*. Po rozpoczęciu wpisywania rozwijana lista pokaże pasujące pozycje ograniczone do tych należących do wybranych *Hostów* lub hostów w wybranych *Grupach hostów*, jeśli takie wybrano. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można na przykład dodać indywidualnie pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (np. item2, item3). Podczas konfigurowania widżetu na pulpicie szablonu ten parametr pozwala wybierać tylko pozycje skonfigurowane w szablonie.
Tagi pozycji	Określ tagi, aby ograniczyć liczbę pozycji wyświetlanych w widżecie. Więcej informacji znajduje się powyżej w sekcji *Tagi hostów*.
Pokaż hosty w utrzymaniu	Zaznacz to pole wyboru, aby wyświetlać hosty w utrzymaniu (w takim przypadku obok nazwy hosta będzie wyświetlana ikona utrzymania). Ten parametr ma etykietę *Pokaż dane w utrzymaniu* podczas konfigurowania widżetu na pulpicie szablonu.
Pokaż	Zaznacz to pole wyboru, aby wyświetlić odpowiedni element komórki honeycomb — etykietę główną, etykietę pomocniczą. Co najmniej jeden element musi być wybrany.
Konfiguracja zaawansowana	Kliknij nagłówek *Konfiguracja zaawansowana*, aby wyświetlić opcje konfiguracji zaawansowanej.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana* i tylko dla elementów wybranych w polu *Pokaż* (zobacz wyżej) oraz dla koloru tła lub progów komórek plastra miodu.



Etykieta podstawowa/drugorzędna

Typ — Wybierz typ etykiety:
Tekst — etykieta będzie wyświetlać tekst określony dla parametru *Tekst*;
Wartość — etykieta będzie wyświetlać wartość pozycji z liczbą miejsc po przecinku określoną dla parametru *Miejsca po przecinku*.

Tekst — Wprowadź tekst etykiety. Ten tekst może zastąpić domyślną nazwę pozycji.
Obsługiwany jest tekst wielowierszowy. Możliwe jest połączenie tekstu i obsługiwanych makr.
Obsługiwane są `{HOST.\*}`, `{ITEM.\*}`, `{INVENTORY.\*}` oraz makra użytkownika.
Komórki plastra miodu są porządkowane alfabetycznie według nazwy hosta, a w obrębie każdego hosta — według nazwy pozycji.
Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Tekst*.

Miejsca po przecinku — Wprowadź liczbę miejsc po przecinku wyświetlanych wraz z wartością.
Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Wartość*, i ma wpływ tylko na pozycje zwracające dane numeryczne (float).

Rozmiar — Wybierz rozmiar etykiety:
Automatycznie — użyj automatycznie dopasowanego rozmiaru etykiety;
Niestandardowy — wprowadź niestandardowy rozmiar etykiety (w procentach, względem rozmiaru komórki plastra miodu).
Zwróć uwagę, że etykiety, które nie mieszczą się w rozmiarze komórki plastra miodu, są obcinane.

Pogrubienie — Zaznacz pole wyboru, aby wyświetlać etykiety pogrubioną czcionką.

Kolor — Wybierz kolor etykiety z próbnika kolorów.
„D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Jednostki

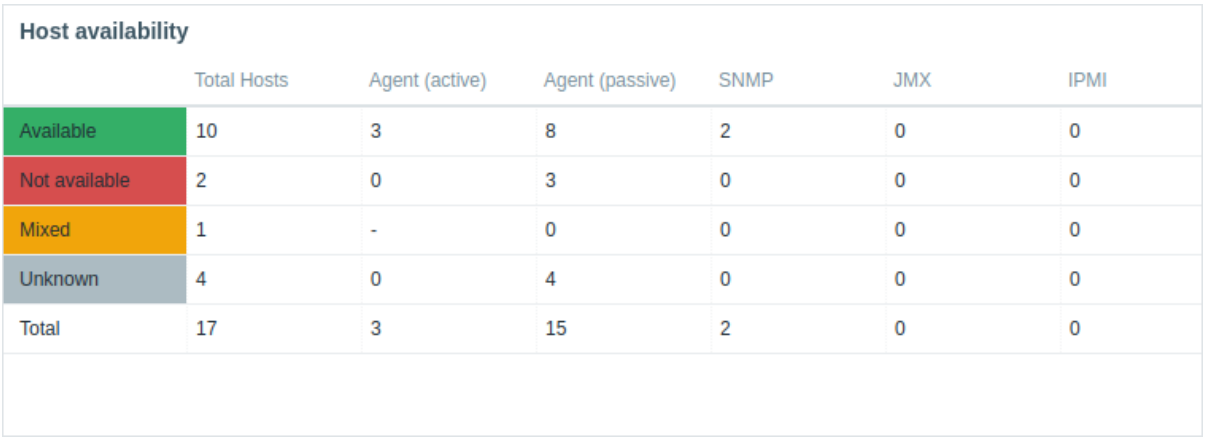
Jednostki — Zaznacz pole wyboru, aby wyświetlać jednostki wraz z wartością pozycji.
Jeśli wprowadzisz nazwę jednostki, zastąpi ona jednostki ustawione w konfiguracji pozycji.
Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Wartość*.

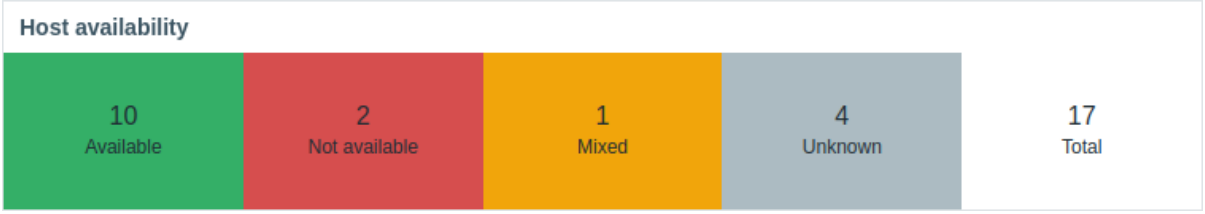
Pozycja	Wybierz położenie jednostek pozycji (przed lub po wartości pozycji). Ten parametr jest ignorowany dla następujących jednostek związanych z czasem: unixtime, uptime, s. Ten parametr jest dostępny, jeśli *Typ* jest ustawiony na *Wartość* i wybrano opcję *Jednostki*.
Kolor tła	
Kolor tła	Wybierz kolor tła komórek plastra miodu z próbnika kolorów. „D” oznacza kolor domyślny, który zależy od motywu frontend. Aby przywrócić kolor domyślny, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Progi	
Interpolacja kolorów	Zaznacz pole wyboru, aby włączyć płynne przejścia między kolorami progów dla komórek plastra miodu. Ten parametr jest dostępny, jeśli ustawiono co najmniej dwa progi.
Próg	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu z próbnika kolorów i określ wartość liczbową. Lista progów zostanie posortowana rosnąco podczas zapisywania. Zwróć uwagę, że kolory skonfigurowane jako progi będą wyświetlane poprawnie tylko dla pozycji numerycznych. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.

Dostępność hosta

Przegląd

Widżet *Dostępność hostów* wyświetla dostępność hostów w wybranych grupach hostów, umożliwiając monitorowanie liczby hostów, które są dostępne lub niedostępne.



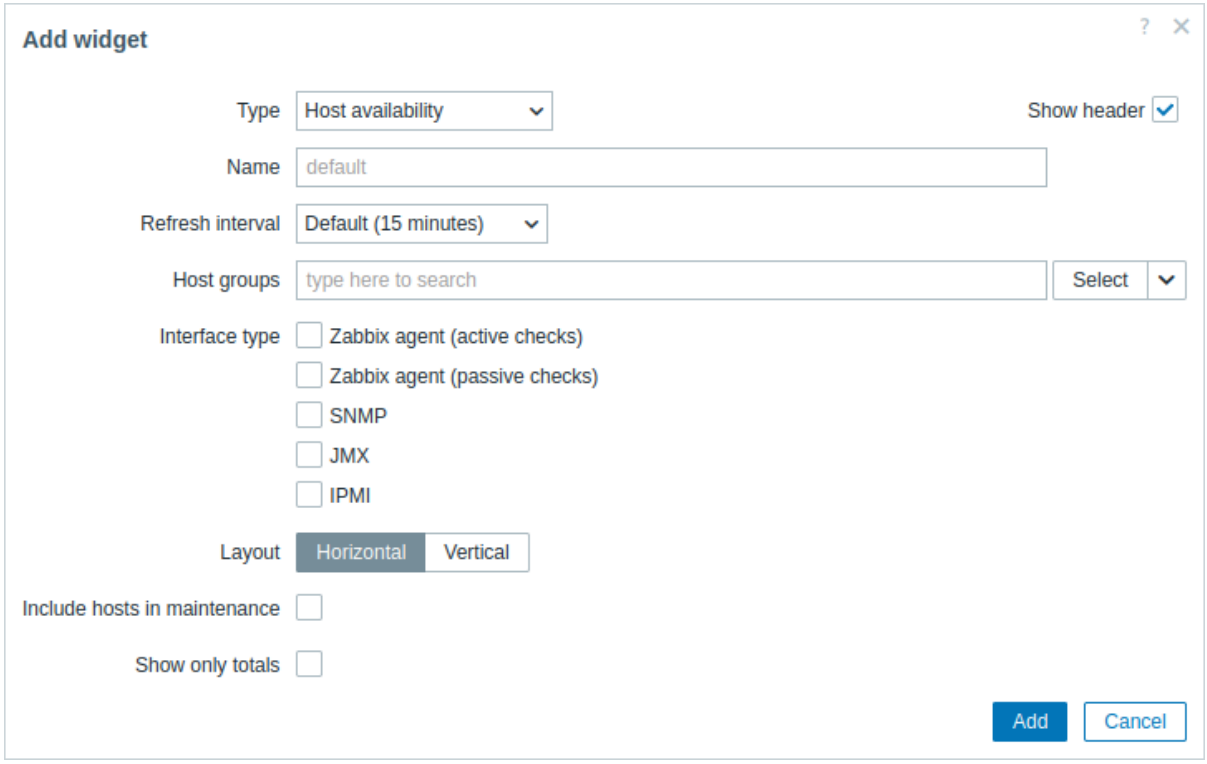


Dostępność hostów jest liczona w następujący sposób:

- *Dostępne* - hosty ze wszystkimi interfejsami dostępnymi
- *Niedostępne* - hosty ze wszystkimi interfejsami niedostępnymi
- *Mieszane* - hosty z co najmniej jednym interfejsem niedostępnym oraz co najmniej jednym dostępnym lub o nieznanym stanie
- *Nieznane* - hosty z co najmniej jednym interfejsem o nieznanym stanie, ale bez żadnego niedostępnego
- *Łącznie* - łączna liczba wszystkich hostów

Konfiguracja

Aby skonfigurować, wybierz *Dostępność hosta* jako typ:



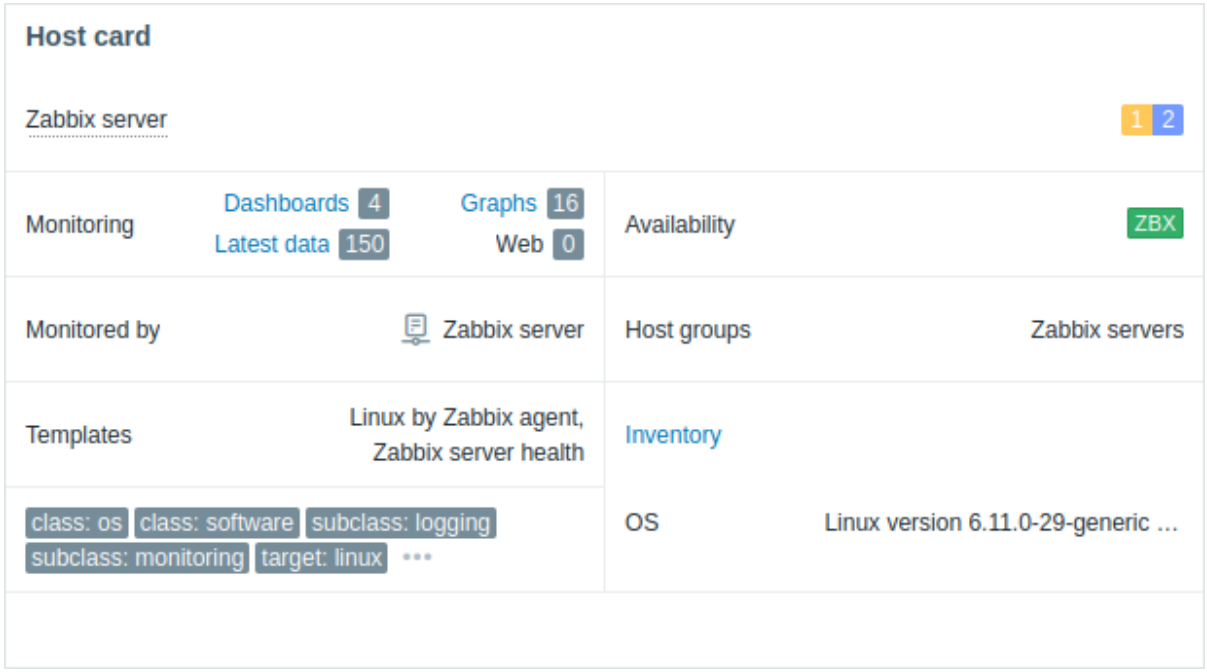
Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz zgodny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Typ interfejsu	Wybierz, dla których interfejsów hosta chcesz wyświetlić dane dostępności — *agent Zabbix (sprawdzenia aktywne)*, *agent Zabbix (sprawdzenia pasywne)*, *SNMP*, *JMX*, *IPMI*. W przypadku *agenta Zabbix (sprawdzenia aktywne)* komórka *Mieszane* będzie zawsze pusta, ponieważ pozycje tego typu nie mogą mieć wielu interfejsów. Jeśli nic nie zostanie wybrane, domyślnie wyświetlana jest dostępność wszystkich interfejsów.
Układ	Wybierz wyświetlanie poziome (kolumny) lub pionowe (wiersze).
Uwzględnij hosty w konserwacji	Uwzględnij w statystykach hosty będące w konserwacji. Ten parametr ma etykietę *Pokaż dane w konserwacji* podczas konfigurowania widgetu na pulpicie szablonu.
Pokaż tylko sumy	Jeśli zaznaczone, wyświetlana jest łączna liczba hostów bez podziału na interfejsy. Ta opcja jest wyłączona, jeśli wybrano tylko jeden interfejs.

Karta hosta

Przegląd

Widżet *Karta hosta* wyświetla szczegóły pojedynczego hosta, umożliwiając szybkie ocenienie stanu i konfiguracji hosta (dostępność, problemy, inwentarz itp.).

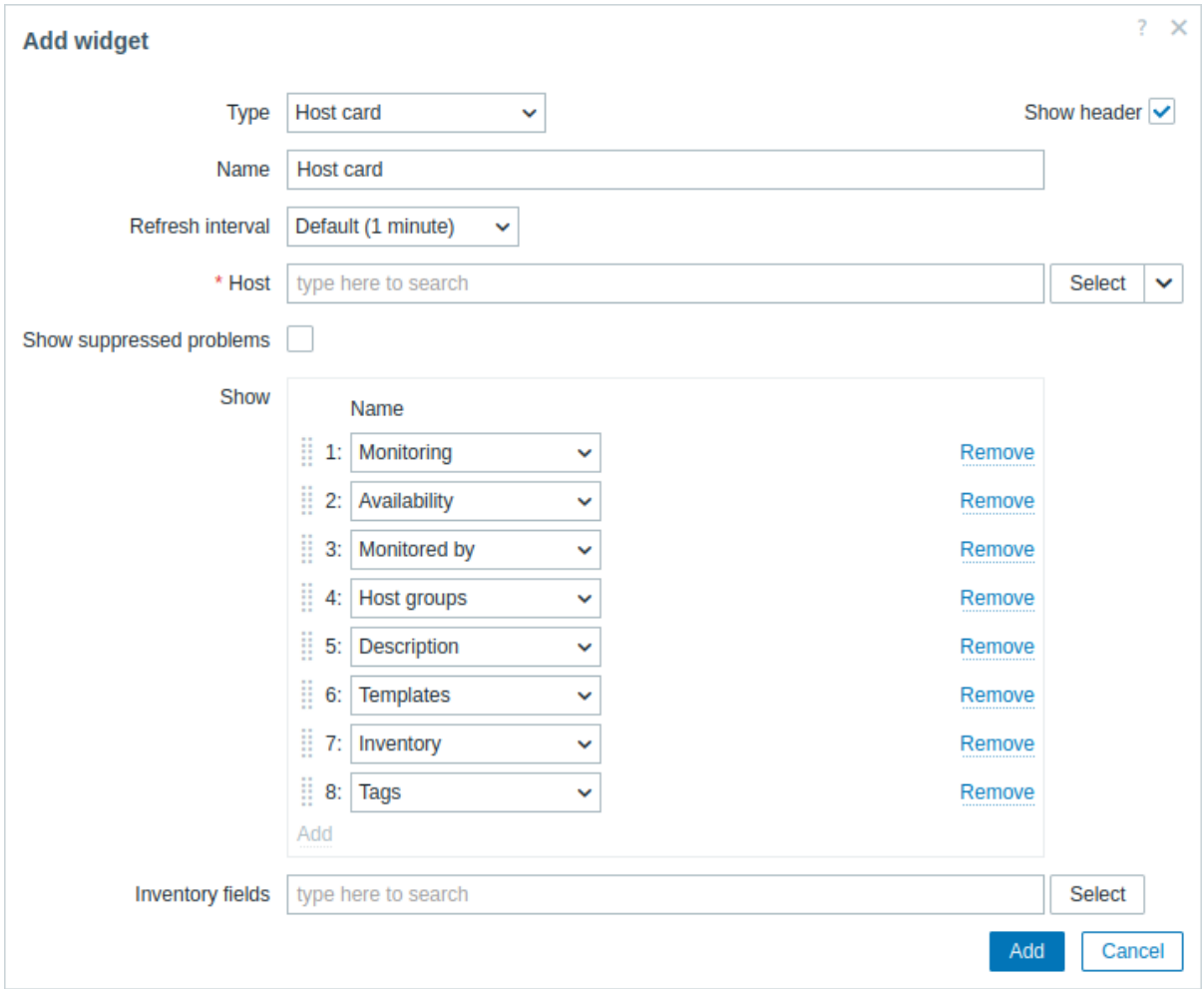


Host może zostać ustawiony bezpośrednio w konfiguracji widżetu lub wybrany z kompatybilnego widżetu albo za pomocą selektora hosta na pulpicie.

Widżet może wyświetlać wiele sekcji, z których każda pokazuje inne informacje. Układ sekcji jest dostosowywany automatycznie na podstawie szerokości widżetu; po rozszerzeniu w poziomie sekcje są automatycznie rozmieszczane w wielu kolumnach.

Konfiguracja

Aby skonfigurować, wybierz *Host card* jako typ:



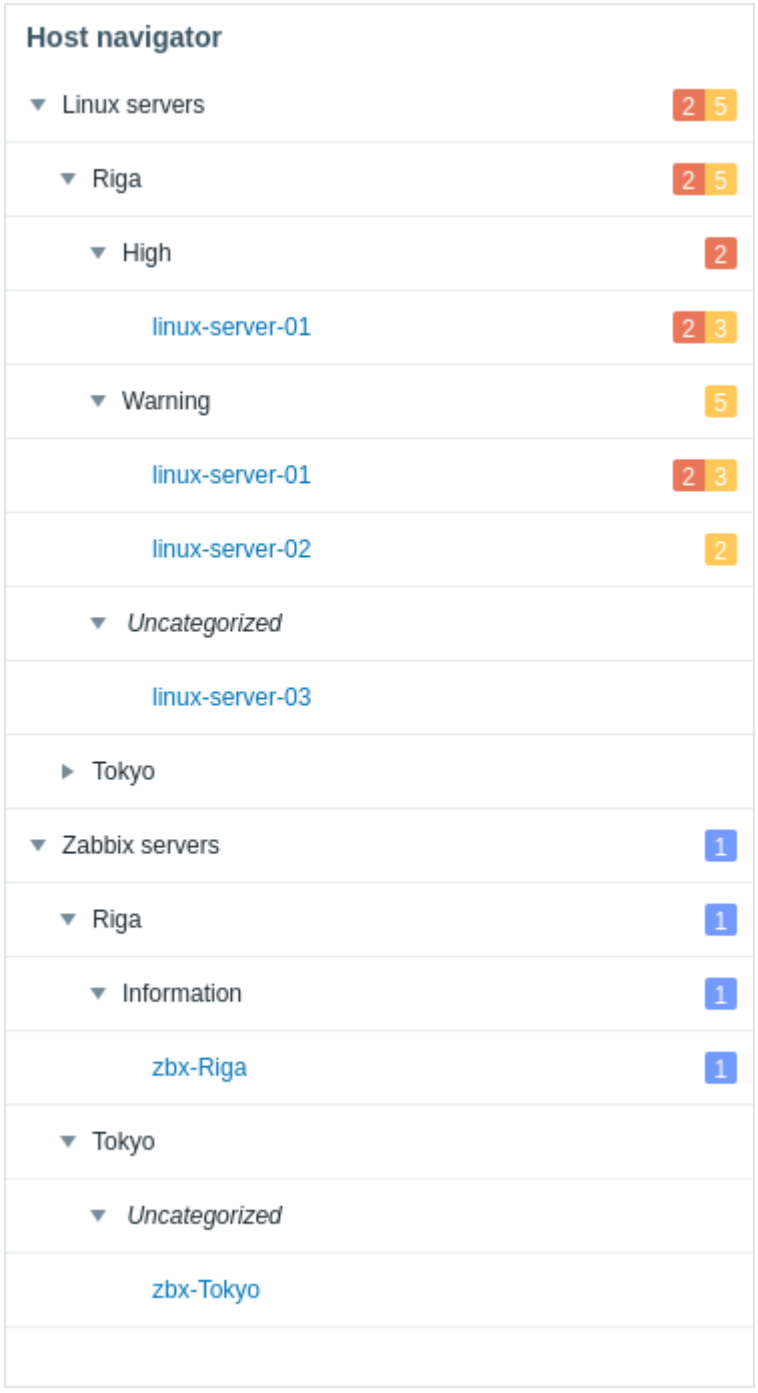
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące parametry specyficzne:

Host	Wybierz host. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. Podczas wpisywania będą wyświetlane pasujące podpowiedzi. Górna sekcja widżetu zawsze wyświetla **nazwę hosta** (kliknięcie otwiera menu hosta): - Jeśli host jest wyłączony, obok nazwy hosta wyświetlana jest czerwona etykieta *Disabled*. - Jeśli host jest w trybie konserwacji, obok nazwy hosta wyświetlana jest ikona konserwacji. Górna sekcja widżetu wyświetla również liczbę **problemów hosta** pogrupowanych według ważności (kliknięcie otwiera problemy hosta). Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Show suppressed problems	Zaznacz to pole wyboru, aby uwzględnić ukryte problemy w obliczaniu problemów.
Show	Dodaj sekcje do wyświetlenia w widżecie. Sekcje można przestawiać metodą przeciągnij i upuść.
Host groups	Wyświetla grupy hostów, do których należy host.
Description	Wyświetla opis hosta.
Monitoring	Wyświetla szybkie odnośniki do przechodzenia do Pulpitów, Najnowszych danych, Wykresów i Scenariuszy WWW hosta. Liczba obok każdego odnośnika wskazuje liczbę odpowiednich encji.
Availability	Wyświetla dostępność hosta według interfejsu.
Monitored by	Wyświetla, czy host jest monitorowany przez serwer Zabbix, czy przez określony proxy lub grupę proxy.
Templates	Wyświetla szablony podłączone do hosta.
Inventory	Wyświetla pola inwentarza hosta.
Tags	Wyświetla tagi hosta.
Inventory fields	Wybierz pola inwentarza do wyświetlenia. Podczas wpisywania będą wyświetlane pasujące podpowiedzi. Jeśli nie określono żadnych pól inwentarza, zostaną wyświetlone wszystkie wypełnione pola inwentarza. Ten parametr jest dostępny, jeśli *Show* zawiera *Inventory*.

Nawigator hostów

Przegląd

Widżet *Host navigator* wyświetla listę hostów na podstawie różnych opcji filtrowania i grupowania.



Grupy, według których hosty są uporządkowane, można rozwijać lub zwijać.

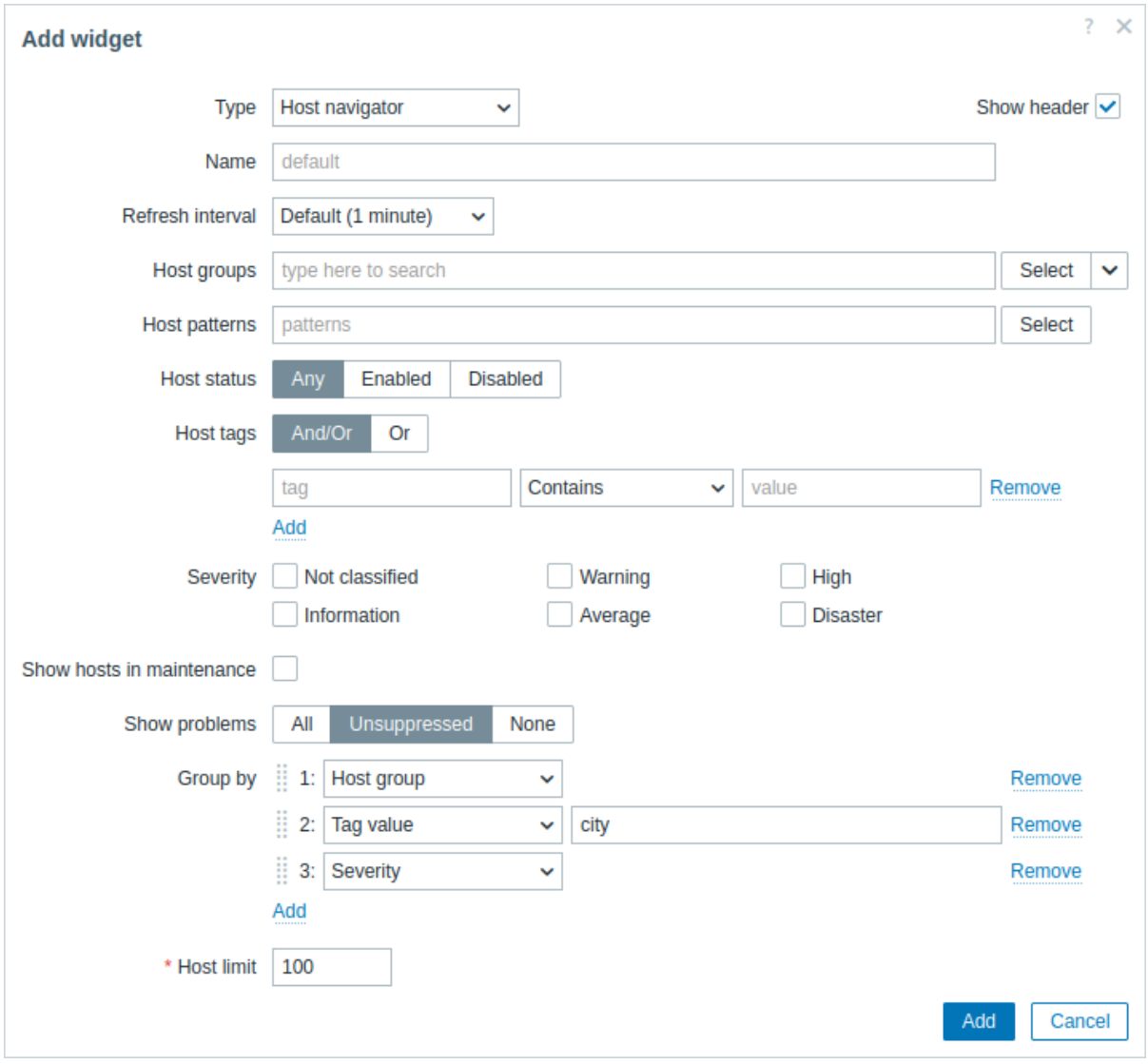
W przypadku grup, problemów oraz hostów w trybie konserwacji dodatkowe szczegóły są dostępne po najechaniu kursorem.

Widżet jest najbardziej przydatny do sterowania tym, co wyświetlają inne widżety, na podstawie wybranego hosta.



Konfiguracja

Aby skonfigurować, wybierz *Host navigator* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje:

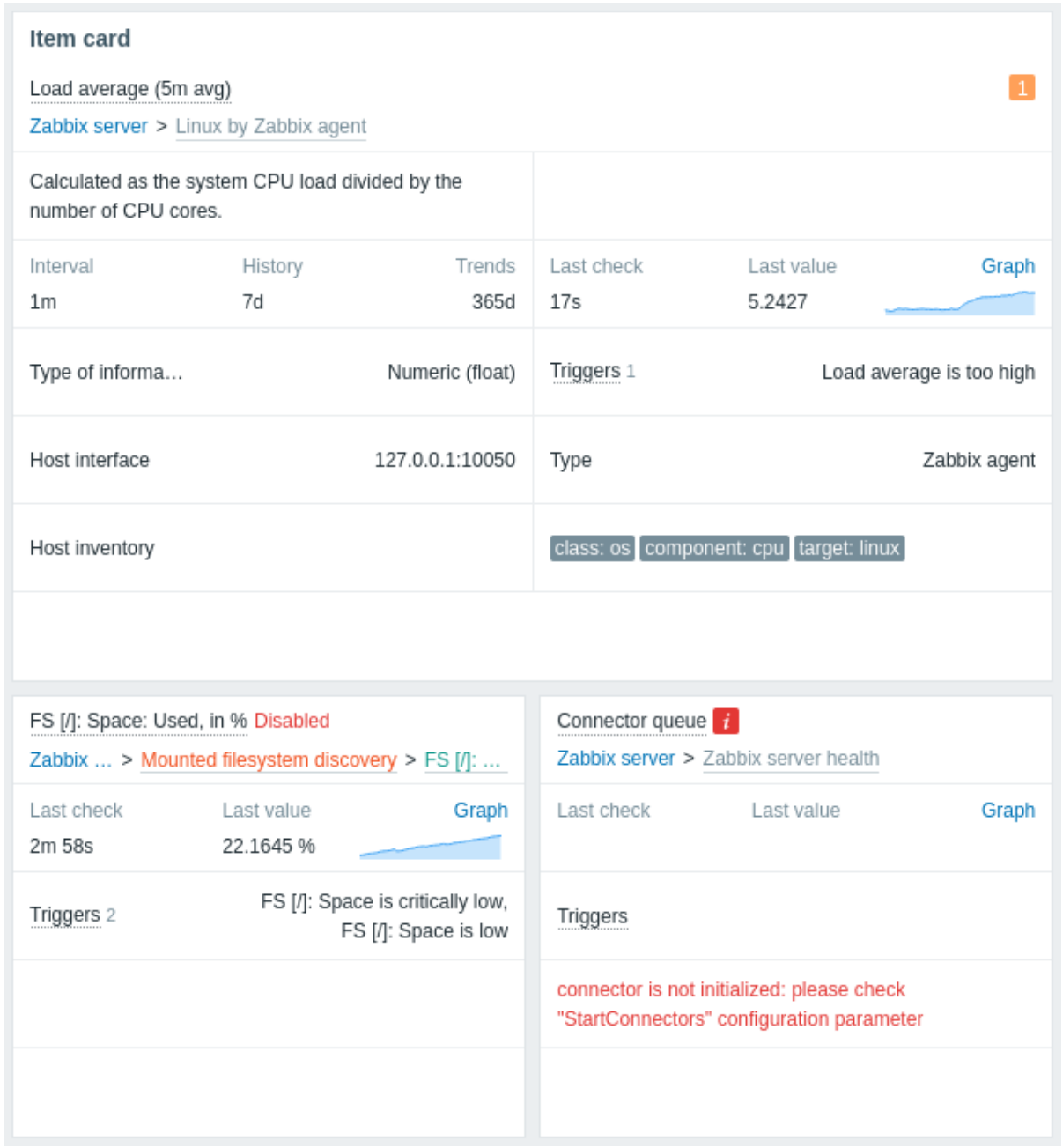
Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz kompatybilny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy wyświetli listę rozwijaną pasujących grup. Wybranie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów; jeśli nie wybrano żadnych grup hostów, widżet wyświetli wszystkie hosty ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Wzorce hostów	Wprowadź wzorce hostów lub wybierz istniejące hosty jako wzorce hostów. Hosty pasujące do określonych wzorców zostaną wyświetlone w nawigatorze hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta wyświetli listę rozwijaną pasujących hostów. Jeśli nie wybrano żadnych hostów, widżet wyświetli wszystkie hosty. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci hosty pasujące do zera lub większej liczby znaków; `Zabbix*` zwróci hosty, których nazwa zaczyna się od „Zabbix”). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij *Enter*. Po rozpoczęciu wpisywania lista rozwijana wyświetli pasujące hosty, ograniczone do hostów należących do hostów w wybranych *Grupach hostów*, jeśli takie wybrano. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Stan hosta	Filtruj wyświetlane hosty na podstawie ich stanu (dowolny, włączony, wyłączony). Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Tagi hostów	Określ tagi do filtrowania hostów wyświetlanych w widżecie. Możliwe jest uwzględnianie oraz wykluczanie określonych tagów i wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Or; **Or** - wystarczy spełnienie jednego warunku. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Ważność	Zaznacz poziomy ważności problemów, aby filtrować hosty z problemami wyświetlanymi w widżecie. Jeśli nie zaznaczono żadnych poziomów ważności, zostaną wyświetlone wszystkie hosty ze wszystkimi problemami.
Pokaż hosty w konserwacji	Zaznacz to pole wyboru, aby wyświetlać hosty w konserwacji (w takim przypadku obok nazwy hosta zostanie wyświetlona ikona konserwacji). Podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu ten parametr nosi nazwę *Pokaż dane w konserwacji*.

Pokaż problemy	Filtruj wyświetlane problemy na podstawie ich stanu (wszystkie, niepominięte, żadne). Problemy są oznaczane kolorowymi blokami na końcu wiersza hosta. Kolory są oparte na ważności problemu, którą można dostosować w sekcji *Administracja* > *Ogólne* > *Opcje wyświetlania wyzwalaczy*. Poziomy ważności problemów można zmieniać podczas aktualizowania problemów.
Grupuj według	Dodaj atrybut grupowania, według którego zostaną pogrupowane wybrane hosty: **Grupa hostów** - grupuj hosty według ich grupy hostów; **Wartość tagu** - wprowadź nazwę tagu, aby grupować hosty według wartości tego tagu (na przykład wprowadź „city”, aby grupować hosty według wartości „Riga”, „Tokyo” itd.); **Ważność** - grupuj hosty według ważności ich problemów. Jeśli opcja *Pokaż problemy* jest skonfigurowana do wyświetlania problemów, są one wyświetlane w następujący sposób: - dla każdej grupy ważności wyświetlana jest tylko odpowiednia liczba problemów; - dla każdego hosta wyświetlane są wszystkie liczby jego problemów. Należy pamiętać, że hosty będą grupowane tylko według poziomów ważności zaznaczonych w parametrze *Ważność*; jeśli nie zaznaczono żadnych poziomów ważności, wszystkie hosty zostaną pogrupowane według wszystkich poziomów ważności. Atrybuty grupowania można zmieniać kolejnością, przeciągając je w górę lub w dół za uchwyt znajdujący się przed nazwą grupy. Należy pamiętać, że kolejność atrybutów grupowania określa kolejność zagnieżdżania grup. Na przykład określenie wielu nazw tagów (1: color, 2: city) spowoduje pogrupowanie hostów według koloru (red, blue itd.), a następnie według miasta (Riga, Tokyo itd.). Host może być wyświetlany w wielu grupach w zależności od skonfigurowanych atrybutów grupowania (na przykład podczas grupowania według grupy hostów, gdy host należy do wielu grup hostów). Kliknięcie takich hostów zaznacza je i wyróżnia we wszystkich grupach. Hosty, które nie pasują do skonfigurowanych atrybutów grupowania, są wyświetlane w grupie *Nieskategoryzowane*. Można określić maksymalnie 10 atrybutów grupowania, a wszystkie muszą być unikatowe. Jeśli nie określono żadnych atrybutów grupowania, hosty nie będą grupowane.
Limit hostów	Wprowadź maksymalną liczbę hostów, które mają zostać wyświetlone. Możliwe wartości mieszczą się w zakresie od 1 do 9999. Jeśli dostępnych jest więcej hostów niż ustawiony limit, pod wyświetlonymi hostami zostanie pokazany odpowiedni komunikat (na przykład „Wyświetlono 100 z ponad 100 hostów”). Należy pamiętać, że skonfigurowany limit hostów wpływa również na wyświetlanie skonfigurowanych grup; na przykład jeśli limit hostów jest ustawiony na 100, a hosty są grupowane według wartości tagów (więcej niż 200), w widżecie zostanie wyświetlonych tylko pierwszych 100 wartości tagów wraz z odpowiadającymi im hostami. Na ten parametr nie wpływa parametr *Limit wyników wyszukiwania i filtrowania* w sekcji *Administracja* > *Ogólne* > *GUI*. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.

Karta pozycji

Przegląd

Widżet *Karta pozycji* wyświetla szczegóły pojedynczej pozycji, umożliwiając szybkie sprawdzenie stanu i konfiguracji pozycji (najnowsze dane, błędy, opis itp.).

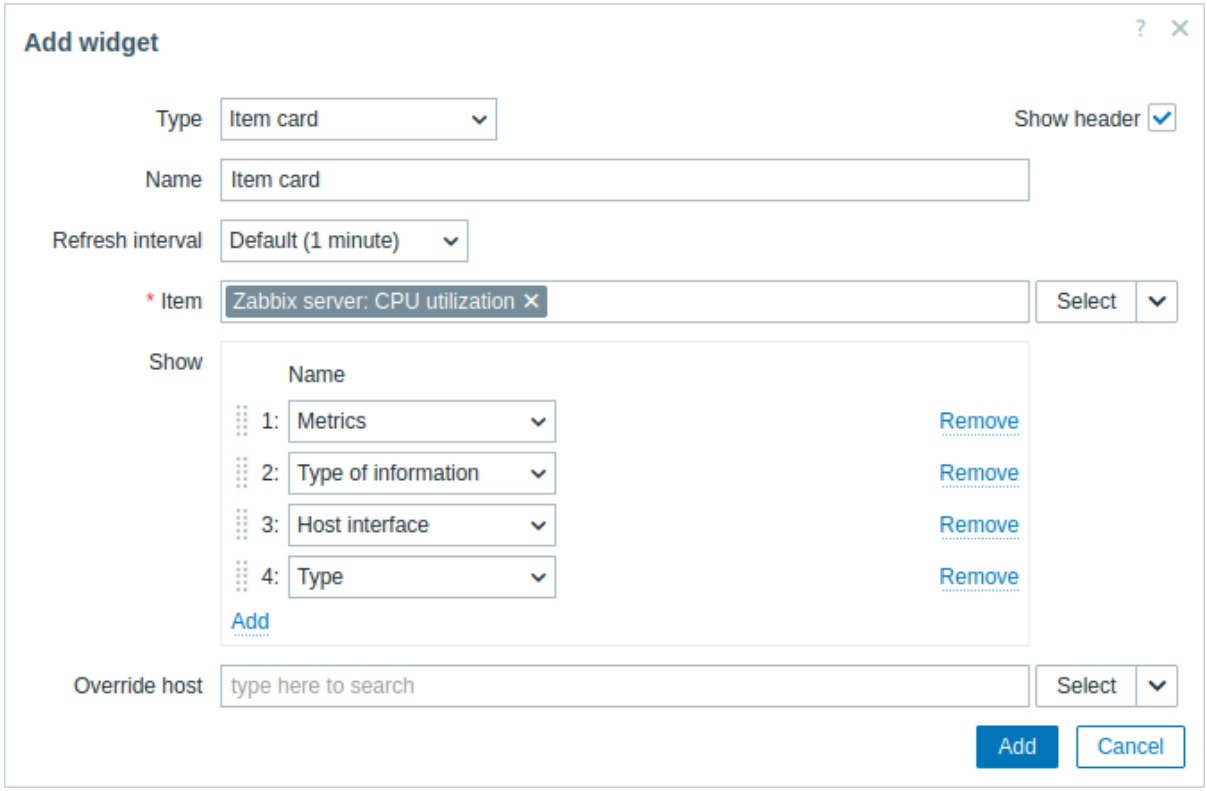


Pozycja może zostać ustawiona bezpośrednio w konfiguracji widżetu lub wybrana z kompatybilnego widżetu na pulpicie.

Widżet może wyświetlać wiele sekcji, z których każda pokazuje inne informacje. Układ sekcji jest dostosowywany automatycznie w zależności od szerokości widżetu; po rozszerzeniu w poziomie sekcje są automatycznie rozmieszczane w wielu kolumnach.

Konfiguracja

Aby skonfigurować widget, wybierz *Item card* jako typ:



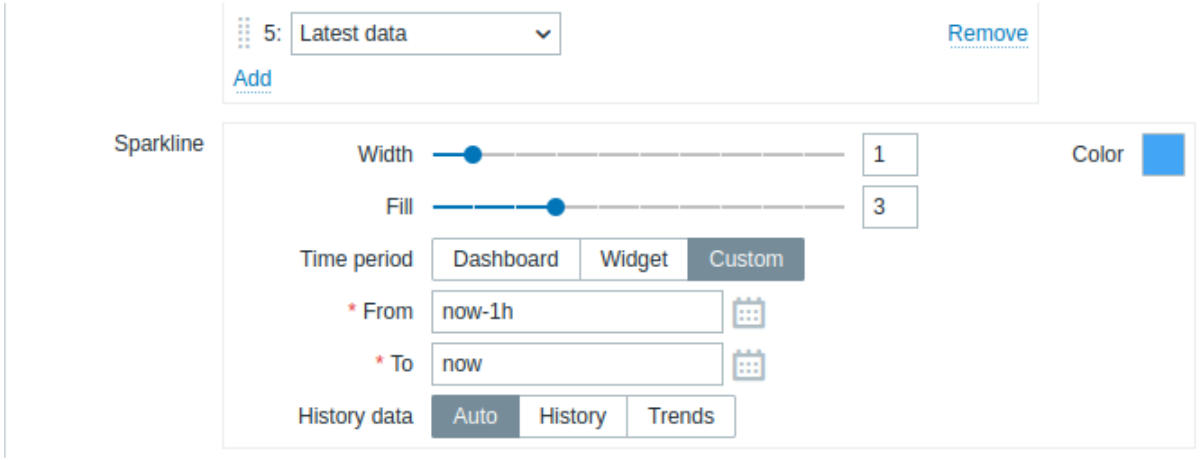
Oprócz parametrów wspólnych dla wszystkich widgetów możesz ustawić następujące opcje specyficzne:

Item	Wybierz pozycję. Alternatywnie wybierz zgodny widget jako źródło danych dla pozycji. Podczas wpisywania wyświetlane są pasujące podpowiedzi. Górna sekcja widgetu zawsze wyświetla **nazwę pozycji** (kliknięcie otwiera menu pozycji): - Jeśli pozycja jest wyłączona, obok nazwy pozycji wyświetlana jest czerwona etykieta *Disabled*. - W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i”. W przypadku problemów dla każdego poziomu ważności problemu wyświetlana jest kwadratowa ikona z liczbą problemów. Najechanie kursorem na ikonę otwiera wyskakujące okno ze szczegółami. - Na pulpicie szablonu wyświetlana jest techniczna nazwa pozycji (zamiast nazwy widocznej). Górna sekcja widgetu wyświetla również **ścieżkę pozycji** — jej host oraz, jeśli dotyczy, szablon, regułę wykrywania niskopoziomowego (LLD) lub pozycję nadrzędną (kliknięcie encji otwiera jej konfigurację). Ścieżka pozycji jest wyświetlana tylko użytkownikom mającym uprawnienia do odpowiednich encji.
Show	Dodaj sekcje do wyświetlenia w widżecie. Sekcje można przestawiać metodą przeciągnij i upuść.
Description	Wyświetla opis pozycji. Makra są rozwijane.
Error text	Wyświetla błąd pozycji (jeśli występuje).
Metrics	Wyświetla następujące informacje: - Interwał aktualizacji pozycji (z wyjątkiem pozycji trapper); - Konfigurację historii pozycji (jeśli istnieje); - Konfigurację trendów pozycji (jeśli istnieje). Makra są rozwijane. Jeśli pozycja używa niestandardowych interwałów, obok interwału aktualizacji pozycji wyświetlana jest ikona pomocy. Najechanie kursorem na ikonę otwiera wyskakujące okno ze szczegółami niestandardowego interwału.

Latest data	Wyświetla następujące informacje: - Czas od ostatniego sprawdzenia pozycji; - Ostatnią wartość pozycji; - Łącze do prostego wykresu/historii wartości pozycji (jeśli historia lub trendy są przechowywane); - Konfigurowalny wykres sparkline (jeśli pozycja jest numeryczna). Dla pozycji typu binarnego zamiast ostatniej wartości wyświetlana jest miniatura lub opcja *Show*. Najechanie kursorem na miniaturę otwiera wyskakujące okno z obrazem. Najechanie kursorem na *Show* otwiera wyskakujące okno z wartością pozycji (ciąg Base64). Na pulpicie szablonu zamiast ostatniej wartości pozycji wyświetlane jest *No data*.
Type of information	Wyświetla typ informacji pozycji.
Triggers	Wyświetla następujące informacje: - Etykietę *Triggers* z liczbą wyzwalaczy (najechanie kursorem na *Triggers* otwiera wyskakujące okno ze szczegółami wyzwalaczy); - Listę nazw wyzwalaczy (makra są rozwijane).
Host interface	Wyświetla interfejs hosta używany przez pozycję albo *No data*, jeśli pozycja nie ma interfejsu hosta (np. pozycje zależne, proste testy). Na pulpicie szablonu wyświetlane jest *No data*.
Type	Wyświetla typ pozycji.
Host inventory	Wyświetla pole inwentarza hosta, które jest wypełniane wartością pozycji.
Tags	Wyświetla tagi pozycji.
Override host	Wybierz zgodny widget lub selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.

Wykres sparkline

Opcje konfiguracji wykresu sparkline są dostępne tylko wtedy, gdy sekcja *Najnowsze dane* została dodana do widgetu.



Szerokość	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10.
Kolor	Wybierz kolor linii i wypełnienia.
Wypełnienie	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10.
Okres czasu	Wybierz źródło danych dla okresu czasu: **Pulpit** - użyj selektora okresu czasu pulpitu; **Widget** - użyj zgodnego widgetu (ustawionego w parametrze *Widget*); **Niestandardowy** - użyj niestandardowego okresu czasu ustawionego w parametrach *Od* i *Do*; jeśli jest ustawiony, w prawym górnym rogu widgetu będzie wyświetlana ikona zegara, wskazująca ustawiony czas po najechaniu kursorem. Zwróć uwagę, że niezależnie od konfiguracji *Okresu czasu* widgetu, zgodne widgety nadal mogą używać go jako źródła danych dla okresu czasu.
Widget	Wprowadź lub wybierz zgodny widget jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na „Widget”.

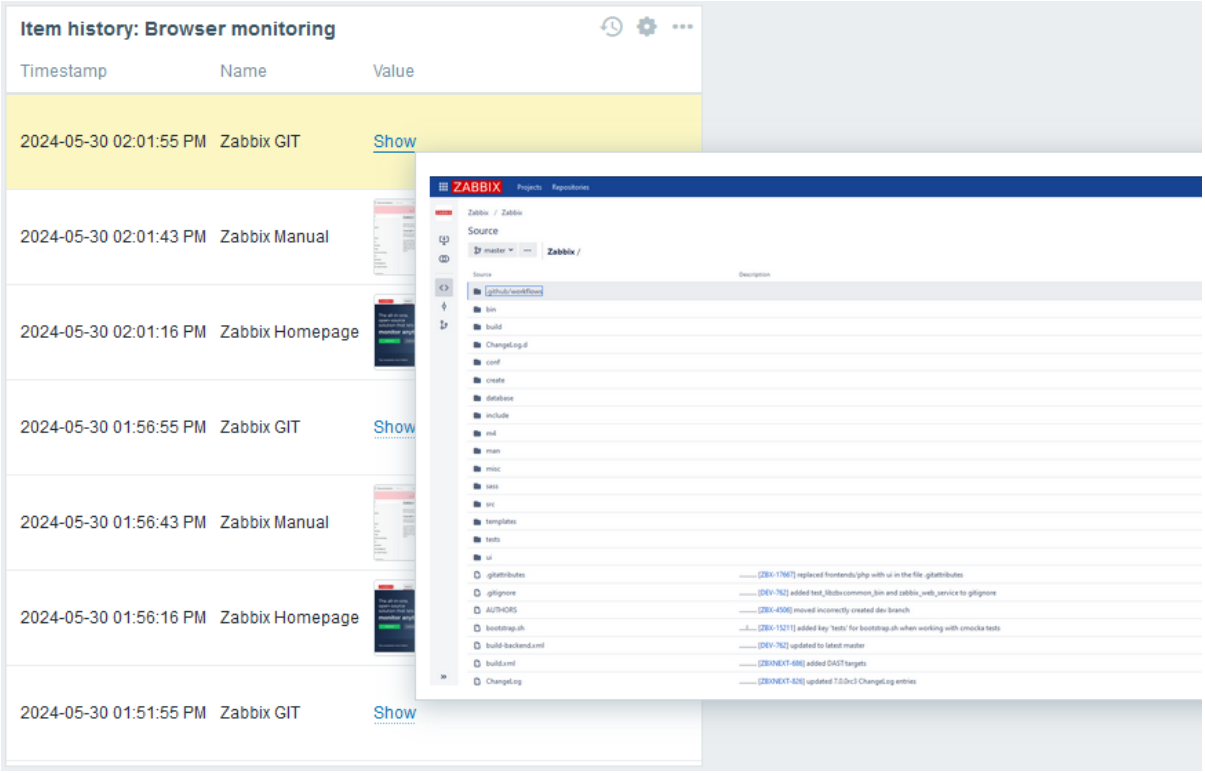
Od	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na „Niestandardowy”.
Dane historyczne	Pobierz dane z historii lub trendów: **Auto** - wybór automatyczny; **Historia** - pobierz dane historyczne; **Trendy** - pobierz dane trendów. Ten parametr ma zastosowanie tylko do danych numerycznych. Dane nienumeryczne będą zawsze pobierane z historii.

Historia pozycji

Przegląd

Widget *Item history* wyświetla najnowsze dane dla wszystkich typów pozycji (numerycznych, tekstowych itp.) w formie tabeli. Może wyświetlać paski postępu i obrazy (przydatne dla pozycji przeglądarkowych) oraz wyróżniać wartości (przydatne dla monitorowania plików dziennika).

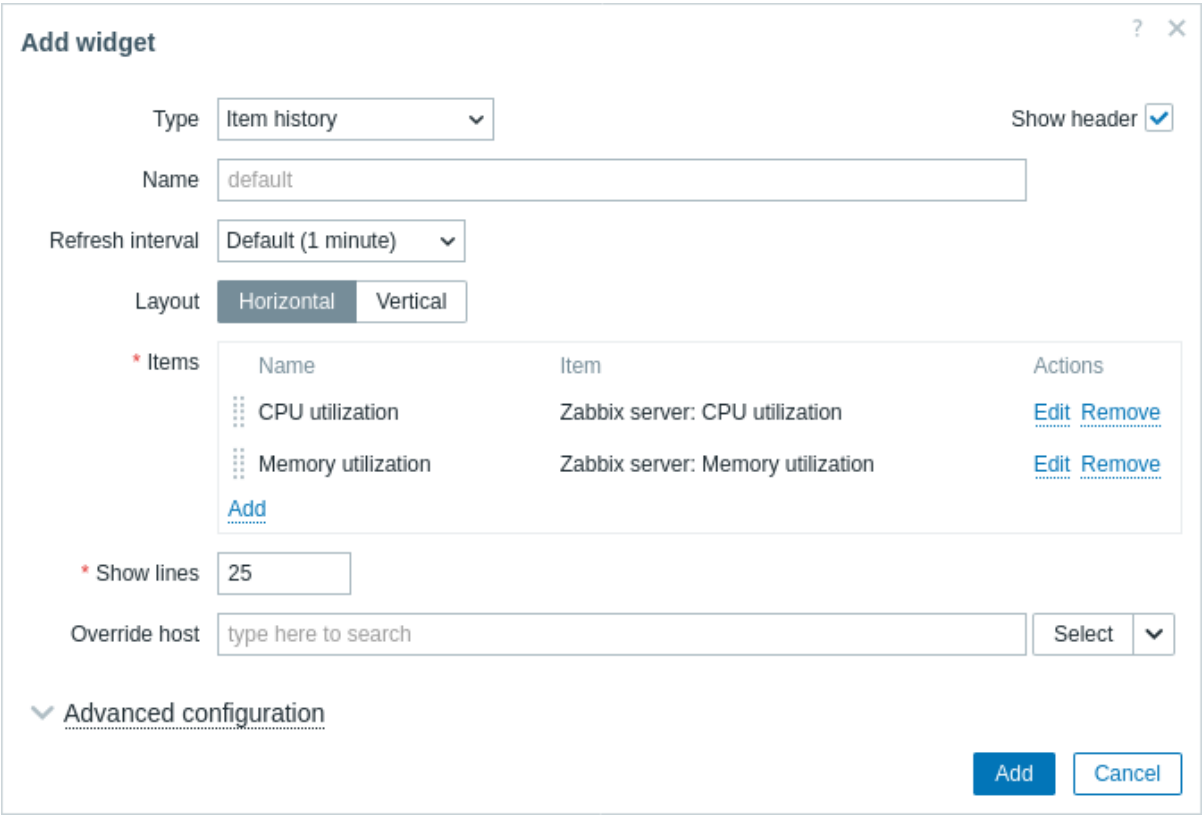




Można wyświetlić do 1000 rekordów.

Konfiguracja

Aby skonfigurować widget, wybierz *Item history* jako typ:



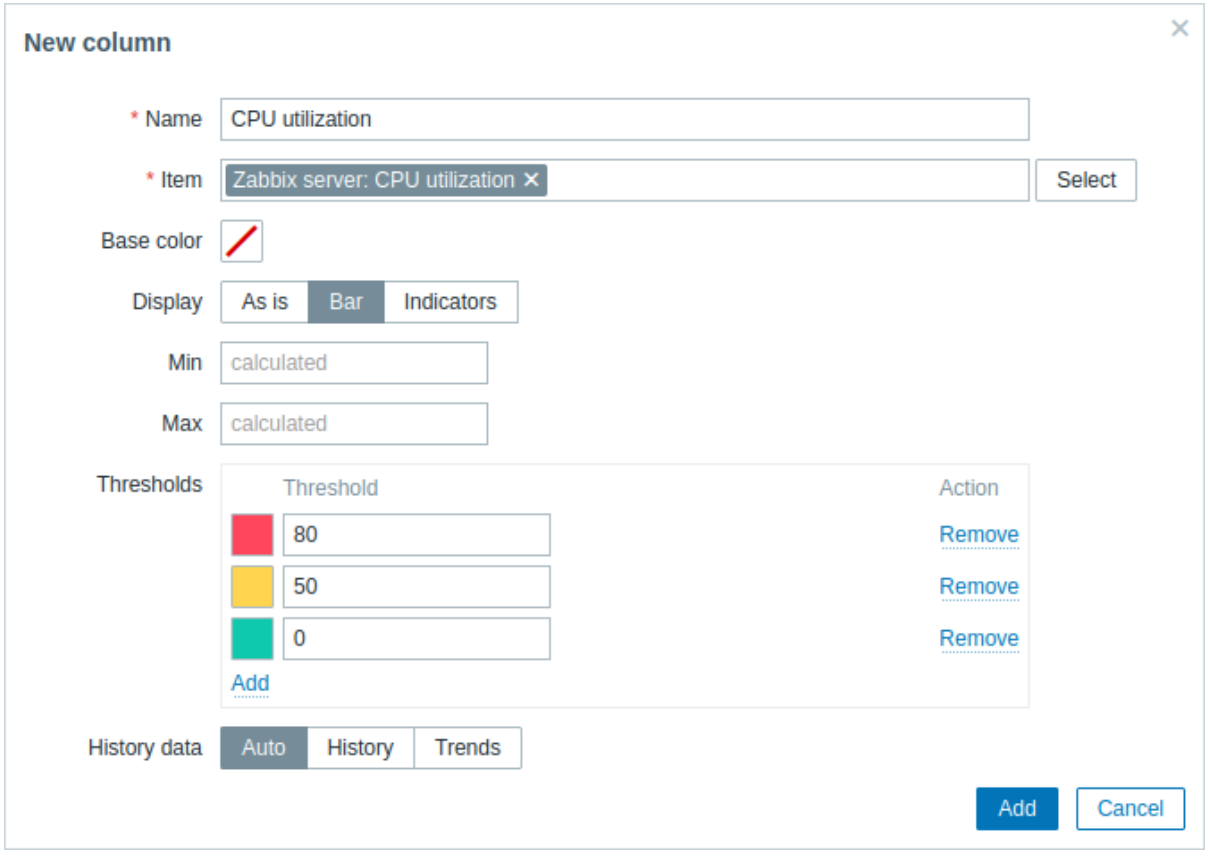
Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Layout	Wybierz układ kolumn pozycji: **Horizontal** — pozycje będą wyświetlane poziomo, wartości pionowo; **Vertical** — pozycje będą wyświetlane pionowo, wartości poziomo.

Items	Dodaj kolumny pozycji do wyświetlenia. Kolejność pozycji określa kolejność ich wyświetlania. Pozycje można zmieniać miejscami, przeciągając je w górę lub w dół za uchwyt znajdujący się przed nazwą pozycji.
Show lines	Określ liczbę wyświetlanych wierszy wartości pozycji.
Override host	Wybierz kompatybilny widget lub selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Advanced configuration	Kliknij etykietę *Advanced configuration*, aby wyświetlić opcje konfiguracji zaawansowanej.

Konfiguracja kolumn

Aby skonfigurować kolumny pozycji, kliknij *Dodaj* w parametrze *Pozycje*:



Wspólne parametry kolumn:

Nazwa	Wprowadź nazwę kolumny. Jeśli pozostawisz puste, zostanie użyta nazwa pozycji z parametru *Pozycja*.
Pozycja	Wybierz pozycję. Zwróć uwagę, że parametry konfiguracji kolumny różnią się w zależności od typu informacji wybranej pozycji; więcej informacji znajdziesz w opisach poszczególnych parametrów poniżej. Podczas konfigurowania widżetu na pulpicie szablonu można wybrać tylko pozycje skonfigurowane w szablonie.
Kolor bazowy	Wybierz kolor tła kolumny lub kolor wypełnienia, jeśli *Wyświetlanie* jest ustawione na „Słupek” lub „Wskaźniki”. Zwróć uwagę, że kolor bazowy może zostać nadpisany przez kolory progów lub wyróżnienia.

Parametry kolumny specyficzne dla pozycji typu numerycznego:

Wyświetlanie	Wybierz sposób wyświetlania wartości pozycji: **As is** - jako zwykły tekst; **Bar** - jako jednolity pasek wypełniony kolorem; **Indicators** - jako segmentowany pasek wypełniony kolorem.

Min	Wprowadź wartość minimalną dla paska/wskaźników. Jeśli pole pozostanie puste, widget użyje minimalnej wartości pozycji. Ten parametr jest dostępny tylko wtedy, gdy opcja *Wyświetlanie* jest ustawiona na „Bar” lub „Indicators”.
Max	Wprowadź wartość maksymalną dla paska/wskaźników. Jeśli pole pozostanie puste, widget użyje maksymalnej wartości pozycji. Ten parametr jest dostępny tylko wtedy, gdy opcja *Wyświetlanie* jest ustawiona na „Bar” lub „Indicators”.
Progi	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu z próbnika kolorów i określ wartość numeryczną. Lista progów zostanie posortowana rosnąco podczas zapisywania. Obsługiwane są sufiksy (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.
Dane historyczne	Wybierz, czy pobierać dane z historii, czy z trendów: **Auto** - wybór automatyczny; **History** - pobierz dane historyczne; **Trends** - pobierz dane trendów.

Parametry kolumn specyficzne dla pozycji typu character, text, log i JSON:

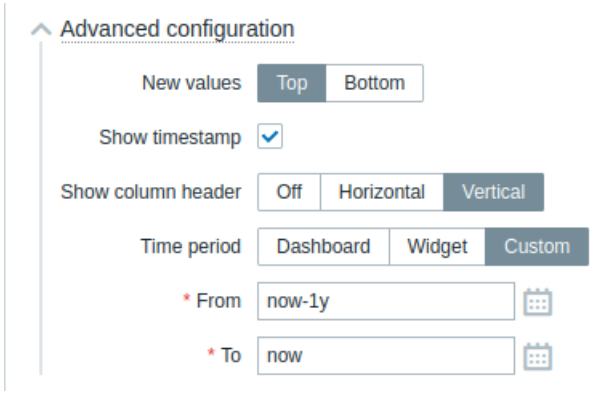
Highlights	Kliknij *Add*, aby dodać highlight, wybierz kolor highlightu z selektora kolorów i określ wyrażenie regularne. Wybrany kolor będzie używany jako kolor tła dla wartości pozycji, w których określone wyrażenie regularne pasuje do tekstu.
Display	Wybierz sposób wyświetlania wartości pozycji: **As is** - dokładnie tak, jak została odebrana. Podziały wierszy są zachowywane, jeśli wartość zawiera znaki końca wiersza. Zawijanie wyrazów jest stosowane, jeśli *Layout* jest ustawione na ”Vertical” i skonfigurowano więcej niż jedną kolumnę; **HTML** - jako tekst sformatowany w HTML; **Single line** - jako pojedyncza linia, skrócona do określonej długości (1-500 znaków). Najazd kursorem lub kliknięcie skróconej wartości otwiera okno podręczne z pełną wartością.
Use monospace font	Zaznacz to pole wyboru, aby wyświetlać wartość pozycji czcionką monospace (domyślnie niezaznaczone).
Display local time	Zaznacz to pole wyboru, aby wyświetlać czas lokalny zamiast znacznika czasu w kolumnie znacznika czasu. Pamiętaj, że pole wyboru *Show timestamp* w advanced configuration również musi być zaznaczone. Ten parametr jest dostępny tylko dla pozycji typu log.

Parametry kolumny specyficzne dla pozycji typu binarnego:

Pokaż miniaturę	Zaznacz to pole wyboru, aby wyświetlać miniaturę dla binariów obrazów lub opcję „Pokaż” dla binariów niebędących obrazami. Odznacz to pole wyboru, aby wyświetlać opcję „Pokaż” dla wszystkich wartości binarnych pozycji. Najechanie kursorem na opcję „Pokaż” lub kliknięcie jej otwiera wyskakujące okno z wartością pozycji (obrazem lub ciągiem Base64). Jeśli wartość pozycji jest pustym ciągiem, wyświetlana jest opcja „Pokaż”; najechanie kursorem na nią lub kliknięcie jej otwiera wyskakujące okno zawierające „Pusty ciąg”.

Konfiguracja zaawansowana

Zaawansowane opcje konfiguracji są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*:

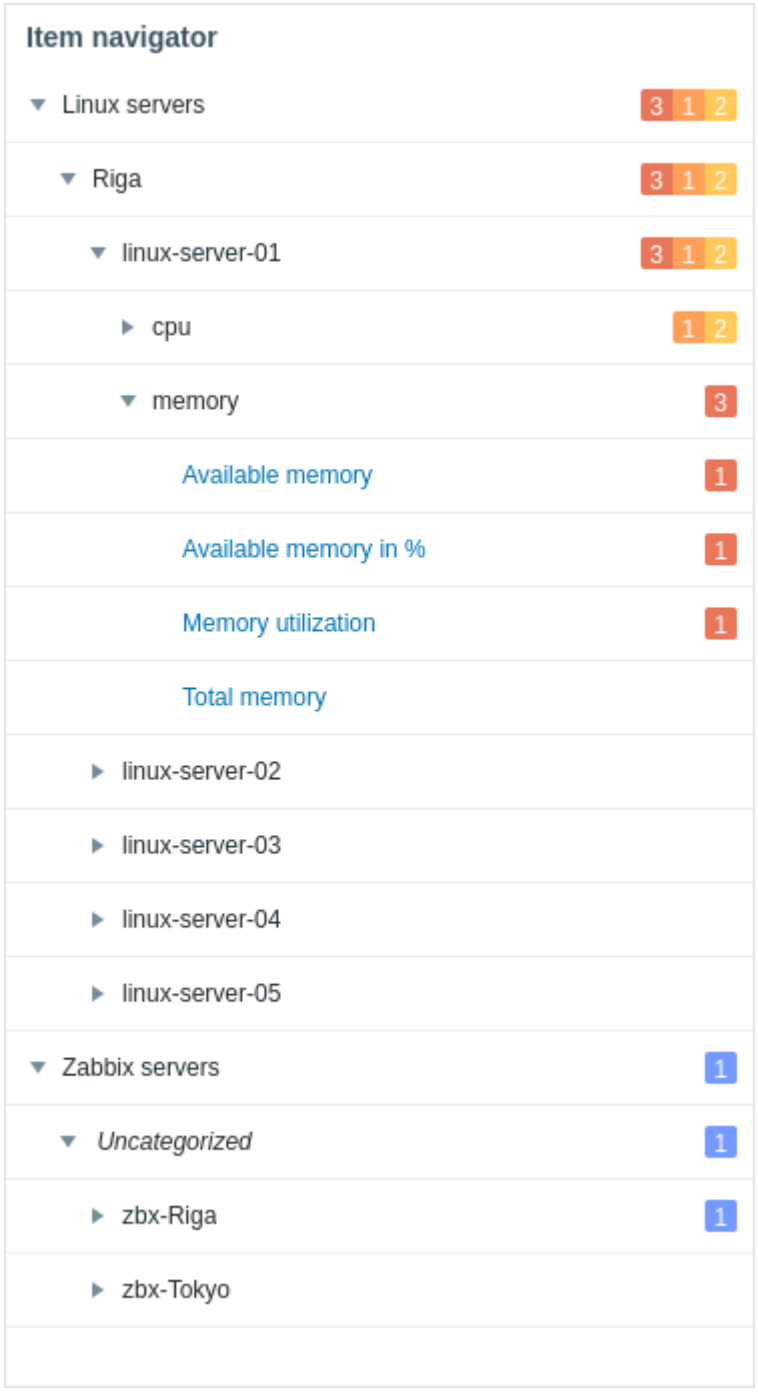


Nowe wartości	Wybierz, gdzie mają być dodawane nowe wartości pozycji: **U góry** - na górze kolumn; **Na dole** - na dole kolumn.
Pokaż znacznik czasu	Zaznacz to pole wyboru, aby wyświetlić kolumnę znacznika czasu (domyślnie niezaznaczone).
Pokaż nagłówek kolumny	Wybierz orientację nagłówka kolumny: **Wyłączone** - ukryj nagłówek; **Poziomo** - wyświetl nagłówek poziomo; **Pionowo** - wyświetl nagłówek pionowo.
Okres czasu	Wybierz źródło danych dla okresu czasu: **Pulpit** - użyj selektora okresu czasu pulpitu; **Widżet** - użyj zgodnego widżetu (ustawionego w parametrze *Widżet*); **Niestandardowy** - użyj niestandardowego okresu czasu ustawionego w parametrach *Od* i *Do*; jeśli jest ustawiony, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem. Zwróć uwagę, że niezależnie od konfiguracji *Okresu czasu* widżetu, zgodne widżety nadal mogą używać go jako źródła danych dla okresu czasu.
Widżet	Wprowadź lub wybierz zgodny widżet (*Graph*, *Graph (classic)*, *Graph prototype*) jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na „Widżet”.
Od	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Okres czasu* jest ustawiony na „Niestandardowy”.

Nawigator pozycji

Przegląd

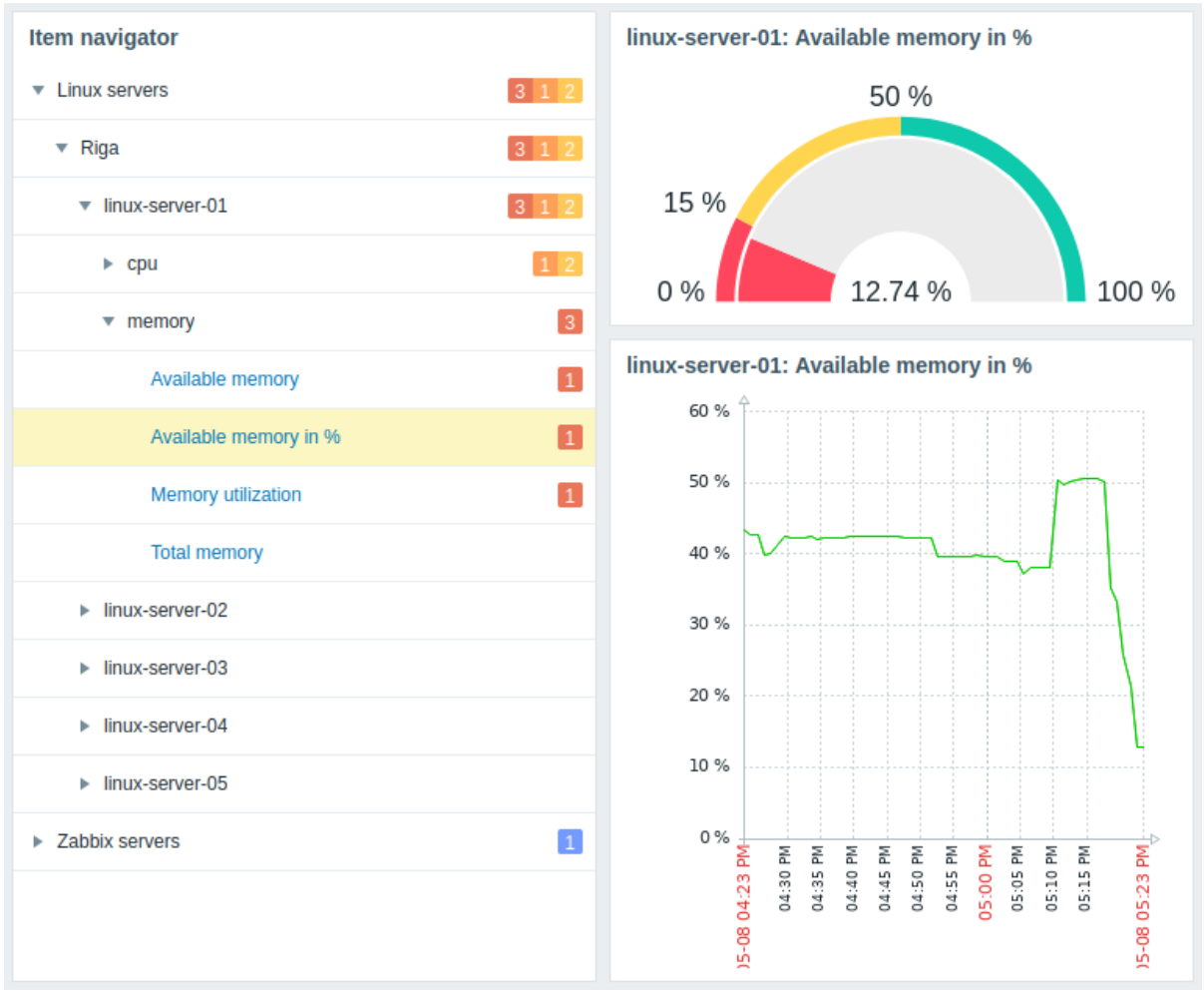
Widżet *Nawigator pozycji* wyświetla listę pozycji na podstawie różnych opcji filtrowania i grupowania.



Grupy, według których pozycje są uporządkowane, można rozwijać lub zwijać.

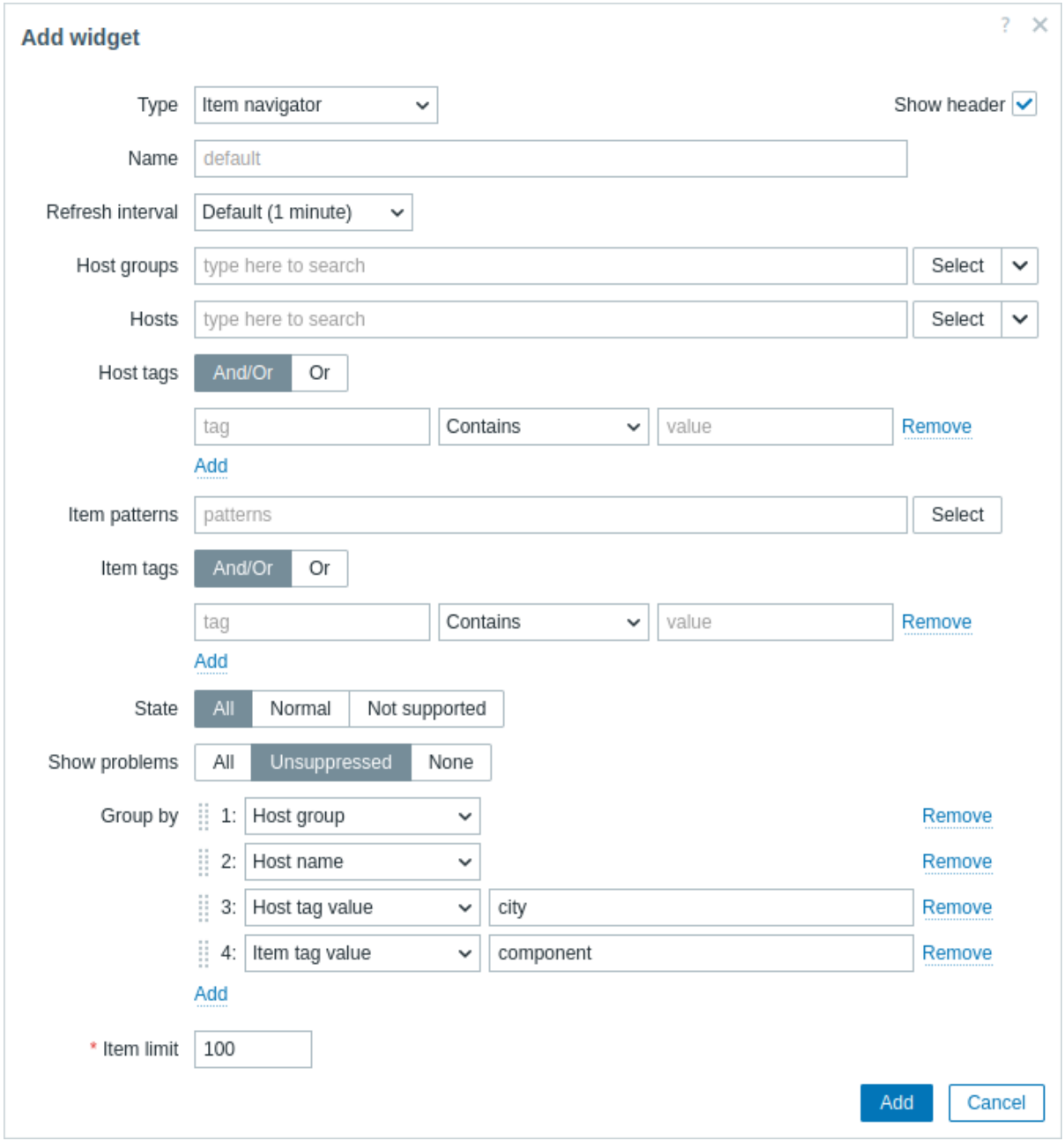
W przypadku grup i problemów dodatkowe szczegóły są dostępne po najechaniu kursorem.

Widżet jest najbardziej przydatny do sterowania tym, co wyświetlają inne widżety, na podstawie wybranej pozycji.



Konfiguracja

Aby skonfigurować, wybierz *Item navigator* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje:

Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz kompatybilny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Wybranie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów; jeśli nie wybrano żadnych grup hostów, widżet wyświetli pozycje należące do wszystkich hostów ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Hosty	Wybierz hosty. Alternatywnie wybierz kompatybilny widżet lub pulpit nawigacyjny jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Jeśli nie wybrano żadnych hostów, widżet wyświetli pozycje należące do wszystkich hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.

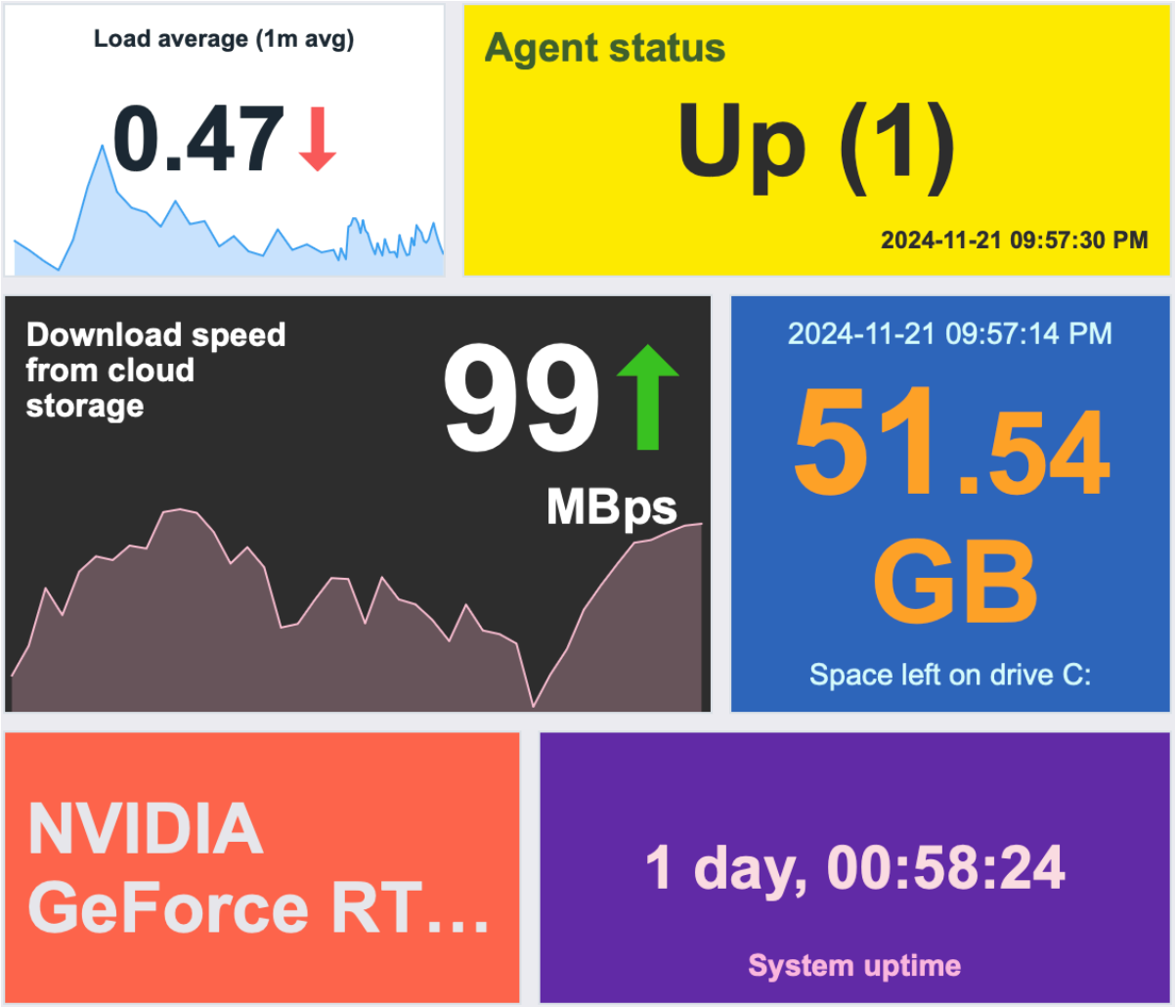
Tagi hosta	Określ tagi hosta, aby filtrować pozycje wyświetlane w widżecie. Możliwe jest uwzględnianie i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Istnieje** - uwzględnia określone nazwy tagów; **Równa się** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter); **Nie istnieje** - wyklucza określone nazwy tagów; **Nie równa się** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Or; **Or** - wystarczy spełnienie jednego warunku. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Wzorce pozycji	Wprowadź wzorce pozycji lub wybierz istniejące pozycje jako wzorce pozycji. Pozycje pasujące do określonych wzorców zostaną wyświetlone w nawigatorze pozycji. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci pozycje pasujące do zera lub większej liczby znaków; `Zabbix*` zwróci pozycje rozpoczynające się od „Zabbix”). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij *Enter*. Po rozpoczęciu wpisywania lista rozwijana wyświetli pasujące pozycje ograniczone do tych należących do wybranych *Hostów* lub hostów w wybranych *Grupach hostów*, jeśli takie wybrano. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać indywidualnie na przykład pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (np. item2, item3). Podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu ten parametr umożliwia wybór wyłącznie pozycji skonfigurowanych w szablonie.
Tagi pozycji	Określ tagi pozycji, aby filtrować pozycje wyświetlane w widżecie. Możliwe jest uwzględnianie i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Istnieje** - uwzględnia określone nazwy tagów; **Równa się** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter); **Nie istnieje** - wyklucza określone nazwy tagów; **Nie równa się** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Or; **Or** - wystarczy spełnienie jednego warunku.
Stan	Filtruj wyświetlane pozycje na podstawie ich stanu (wszystkie, normalne, nieobsługiwane).
Pokaż problemy	Filtruj problemy do wyświetlenia na podstawie ich statusu (wszystkie, nieukryte, brak). Problemy są oznaczane kolorowymi blokami na końcu wiersza pozycji. Kolory zależą od ważności problemu, którą można dostosować w sekcji *Administracja* > *Ogólne* > *Opcje wyświetlania wyzwalaczy*. Ważność problemów można zmienić podczas aktualizowania problemów.

Grupuj według	Dodaj atrybut grupowania, według którego będą grupowane pozycje: **Grupa hostów** - grupuje pozycje według grup hostów, do których należą ich hosty; **Nazwa hosta** - grupuje pozycje według ich hostów; **Wartość tagu hosta** - wprowadź nazwę tagu, aby grupować pozycje według wartości tego tagu hosta (na przykład wprowadź „city”, aby grupować pozycje według wartości „Riga”, „Tokyo” itd.); **Wartość tagu pozycji** - wprowadź nazwę tagu, aby grupować pozycje według wartości tego tagu pozycji (na przykład wprowadź „component”, aby grupować pozycje według wartości „cpu”, „memory” itd.). Atrybuty grupowania można zmieniać kolejnością, przeciągając je w górę lub w dół za uchwyt znajdujący się przed nazwą grupy. Należy pamiętać, że kolejność atrybutów grupowania określa kolejność zagnieżdżania grup. Na przykład określenie kilku nazw tagów hosta (1: color, 2: city) spowoduje grupowanie pozycji według koloru (red, blue itd.), a następnie według miasta (Riga, Tokyo itd.). Pozycja może być wyświetlana w wielu grupach w zależności od skonfigurowanych atrybutów grupowania (na przykład podczas grupowania według grupy hostów, gdy host pozycji należy do wielu grup hostów). Kliknięcie takiej pozycji wybiera ją i wyróżnia we wszystkich grupach. Pozycje, które nie pasują do skonfigurowanych atrybutów grupowania, są wyświetlane w grupie *Nieskategoryzowane*. Jeśli opcja *Pokaż problemy* jest skonfigurowana tak, aby wyświetlać problemy, są one wyświetlane w następujący sposób: - dla każdej grupy wyświetlana jest łączna liczba problemów podrzędnych pozycji; - dla każdej pozycji wyświetlana jest tylko liczba jej problemów. Można określić maksymalnie 10 atrybutów grupowania, a wszystkie muszą być unikatowe. Jeśli nie określono żadnych atrybutów grupowania, pozycje nie będą grupowane.
Limit pozycji	Wprowadź maksymalną liczbę wyświetlanych pozycji. Możliwe wartości mieszczą się w zakresie od 1 do 9999. Jeśli liczba dostępnych pozycji przekracza ustawiony limit, pod wyświetlanymi pozycjami zostanie pokazany odpowiedni komunikat (na przykład „Wyświetlono 100 z ponad 100 pozycji”). Należy pamiętać, że skonfigurowany limit pozycji wpływa również na wyświetlanie skonfigurowanych grup; na przykład jeśli limit pozycji wynosi 100, a pozycje są grupowane według hostów (z których każdy zawiera 200 pozycji), w widżecie zostanie wyświetlony tylko pierwszy host wraz ze 100 jego pozycjami. Na ten parametr nie wpływa parametr *Limit wyników wyszukiwania i filtrowania* w sekcji *Administracja* > *Ogólne* > *GUI*.

Wartość pozycji

Przegląd

Widget *Item value* wyświetla wartość pojedynczej numerycznej lub tekstowej pozycji. Jest przydatny do monitorowania ważnych metryk, wizualizacji progów oraz wykrywania nagłych zmian w danych.



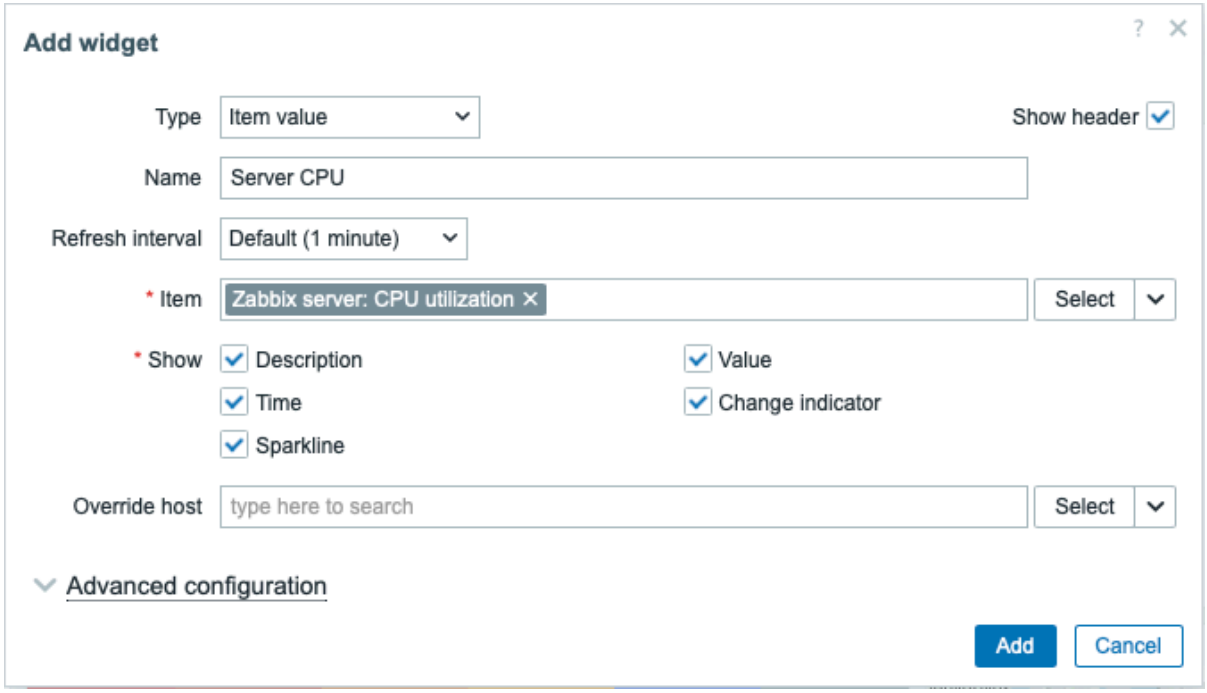
Możesz skonfigurować widget tak, aby wyświetlał następujące elementy:

- Sygnatura czasowa metryki (2024-11-21 09:57:30)
- Opis pozycji (Load average (1m), Agent status)
- Wartość pozycji, wskaźnik zmiany i jednostki (0.47 ↓, 99 ↑ Mbps)
- Kolor tła
- Wykres sparkline dla wartości z określonego przedziału czasu

Kliknięcie widgetu otwiera wykres ad hoc dla pozycji numerycznych lub najnowsze dane dla pozycji tekstowych.

Konfiguracja

Aby skonfigurować widget, wybierz *Item value* jako typ widgetu:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Item	Wybierz pozycję. Alternatywnie wybierz zgodny widget jako źródło danych dla pozycji. Pozycje zwracające dane binarne nie są obsługiwane.
Show	Zaznacz pole wyboru, aby wyświetlić odpowiedni element: opis, wartość, czas, wskaźnik zmiany, wykres przebiegu. Odznacz, aby ukryć. Musi być wybrany co najmniej jeden element.
Override host	Wybierz zgodny widget lub selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić opcje konfiguracji zaawansowanej.

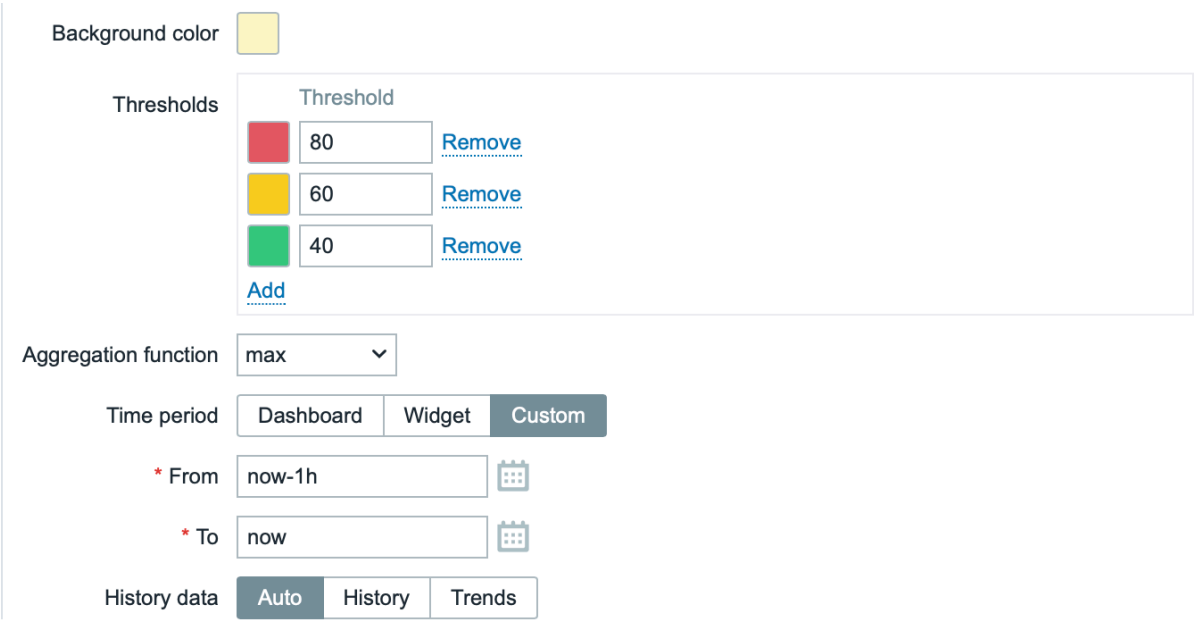
Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*, i tylko dla tych elementów, które są wybrane w polu *Pokaż* (zobacz powyżej).

Należy pamiętać, że wiele elementów nie może zajmować tej samej przestrzeni; jeśli zostaną umieszczone w tej samej przestrzeni, zostanie wyświetlony komunikat o błędzie.

Ogólne parametry widżetu

Te parametry określają kolor tła (statyczny lub dynamiczny) całego widżetu oraz funkcję agregującą używaną do wyświetlania wartości.

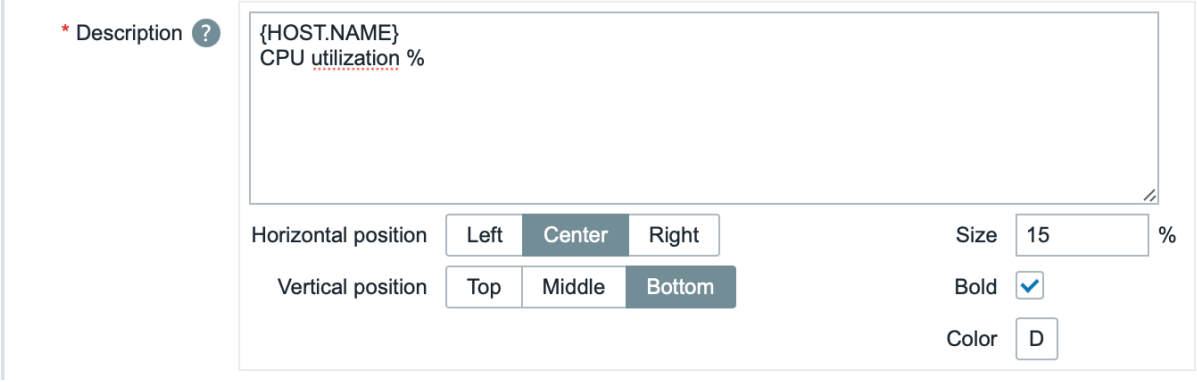


Kolor tła	Wybierz kolor tła całego widżetu za pomocą selektora kolorów. Litera D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w selektorze kolorów.
Progi	Skonfiguruj dynamiczny kolor tła całego widżetu. Kliknij *Dodaj*, aby dodać próg, wybierz kolor tła za pomocą selektora kolorów i określ wartość liczbową. Gdy wartość pozycji będzie równa wartości progowej lub od niej większa, kolor tła ulegnie zmianie. Po zapisaniu lista zostanie posortowana rosnąco. Pamiętaj, że dynamiczny kolor tła będzie wyświetlany prawidłowo tylko dla pozycji liczbowych.
Funkcja agregująca	Określ funkcję agregującą, która ma być używana w wybranym *Okresie*: **min** - wyświetla najmniejszą wartość; **max** - wyświetla największą wartość; **avg** - wyświetla średnią wartość; **count** - wyświetla liczbę wartości; **sum** - wyświetla sumę wartości; **first** - wyświetla pierwszą wartość; **last** - wyświetla ostatnią wartość. Jeśli wybrano opcję **nieużywana**, wyświetlana jest najnowsza wartość (bez agregacji). Agregacja umożliwia wyświetlenie zagregowanej wartości dla określonego przedziału (5 minut, godziny, dnia) zamiast najnowszej wartości. Dla funkcji *min*, *max*, *avg* i *sum* można wyświetlać tylko dane liczbowe. W przypadku funkcji *count* dane nieliczbowe zostaną zmienione na liczbowe.
Okres	Okres używany do agregowania wartości. Wybierz źródło danych dla okresu: **Dashboard** - użyj selektora okresu dashboardu; **Widget** - użyj zgodnego widżetu (ustawionego w parametrze *Widżet*); **Niestandardowy** - użyj niestandardowego okresu ustawionego w parametrach *Od* i *Do*; jeśli jest ustawiony, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara wskazująca ustawiony czas po najechaniu kursorem myszy. Pamiętaj, że niezależnie od konfiguracji parametru *Okres* widżetu zgodne widżety nadal mogą używać go jako źródła danych dla okresu. Ten parametr jest ukryty, gdy dla *Funkcji agregującej* ustawiono wartość „nieużywana”.
Widżet	Wprowadź lub wybierz zgodny widżet jako źródło danych dla okresu. Ten parametr jest dostępny, jeśli dla *Okresu* ustawiono wartość „Widżet”.
Od	Wprowadź lub wybierz początek okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Okresu* ustawiono wartość „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Okresu* ustawiono wartość „Niestandardowy”.

Dane historyczne	Pobieraj dane z historii lub trendów: **Automatycznie** - wybór automatyczny; **Historia** - pobieraj dane historyczne; **Trendy** - pobieraj dane trendów. To ustawienie dotyczy tylko danych liczbowych. Dane nieliczbowe będą zawsze pobierane z historii.

Opis

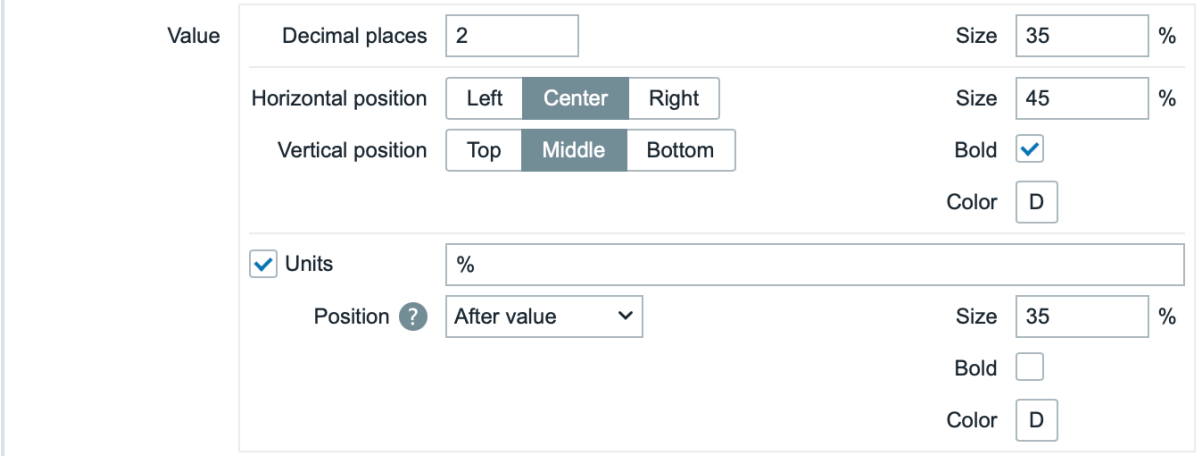
Te parametry określają, w jaki sposób ma być wyświetlany opis pozycji.



Opis	Wprowadź opis pozycji. Ten opis może zastąpić domyślną nazwę pozycji. Obsługiwane są opisy wielowierszowe. Możliwe jest połączenie tekstu i obsługiwanych makr. Obsługiwane są {HOST.*}, {ITEM.*}, {INVENTORY.*} oraz makra użytkownika.
Pozycja pozioma	Wybierz poziome położenie opisu pozycji — z lewej, z prawej lub na środku.
Pozycja pionowa	Wybierz pionowe położenie opisu pozycji — u góry, u dołu lub pośrodku.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla opisu pozycji (w procentach względem całkowitej wysokości widżetu).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać opis pozycji pogrubioną czcionką.
Kolor	Wybierz kolor opisu pozycji z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Wartość

Te parametry określają, jak powinna być wyświetlana wartość pozycji.

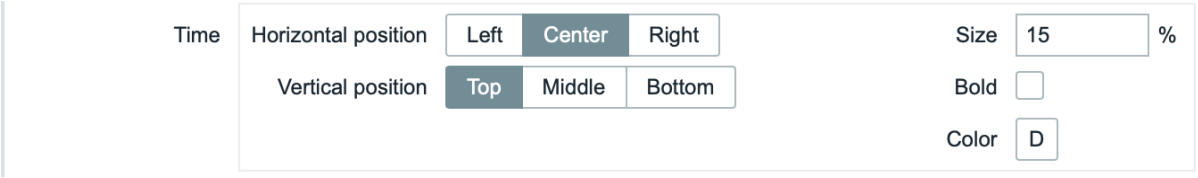


Miejsca dziesiętne	Wybierz, ile miejsc dziesiętnych ma być wyświetlanych wraz z wartością. Ta wartość wpływa tylko na pozycje typu float. Dla pozycji z jednostkami ustawionymi na „s”, jeśli ta wartość jest większa od 0 (domyślnie 2), widżet zaokrągla najbardziej znaczącą jednostkę czasu i wyświetla wartość liczbową (np. „10.43m”) zamiast konwertować wartość na pełny ciąg czasu. Po ustawieniu na 0 sekundy są konwertowane na czytelny dla człowieka ciąg czasu (na przykład „4h 56m 30s”).

Rozmiar	Wprowadź wysokość rozmiaru czcionki dla miejsc dziesiętnych (w procentach względem całkowitej wysokości widżetu).
Pozycja pozioma	Wybierz poziome położenie wartości pozycji — z lewej, z prawej lub na środku.
Pozycja pionowa	Wybierz pionowe położenie wartości pozycji — u góry, na dole lub pośrodku.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla wartości pozycji (w procentach względem całkowitej wysokości widżetu). Zwróć uwagę, że rozmiar wartości pozycji ma priorytet; pozostałe elementy muszą ustąpić jej miejsca. Jednak w przypadku wskaźnika zmiany, jeśli wartość jest zbyt duża, zostanie przycięta, aby pokazać wskaźnik zmiany.
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać wartość pozycji pogrubioną czcionką.
Kolor	Wybierz kolor wartości pozycji z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.
Jednostki	
Jednostki	Zaznacz pole wyboru, aby wyświetlać jednostki wraz z wartością pozycji. Jeśli wprowadzisz nazwę jednostki, zastąpi ona jednostkę z konfiguracji pozycji.
Pozycja	Wybierz położenie jednostki pozycji — nad, pod, przed lub za wartością.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla jednostki pozycji (w procentach względem całkowitej wysokości widżetu).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać jednostkę pozycji pogrubioną czcionką.
Kolor	Wybierz kolor jednostki pozycji z próbnika kolorów. D oznacza kolor domyślny (zależy od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Czas

Te parametry określają, w jaki sposób ma być wyświetlany czas (wartość zegara z historii pozycji).



Pozycja pozioma	Wybierz poziome położenie czasu — po lewej, po prawej lub na środku.
Pozycja pionowa	Wybierz pionowe położenie czasu — u góry, na dole lub pośrodku.
Rozmiar	Wprowadź wysokość rozmiaru czcionki dla czasu (w procentach względem całkowitej wysokości widżetu).
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać czas pogrubioną czcionką.
Kolor	Wybierz kolor czasu z próbnika kolorów. D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Wskaźnik zmiany

W tej sekcji można wybrać kolor wskaźników zmiany za pomocą próbnika kolorów.



Dostępne są następujące wskaźniki zmiany:

- ↑ - wartość pozycji wzrosła (dla pozycji numerycznych)
- ↓ - wartość pozycji spadła (dla pozycji numerycznych)
- ↕ - wartość pozycji uległa zmianie (dla pozycji tekstowych i pozycji z mapowaniem wartości)

Uwaga:

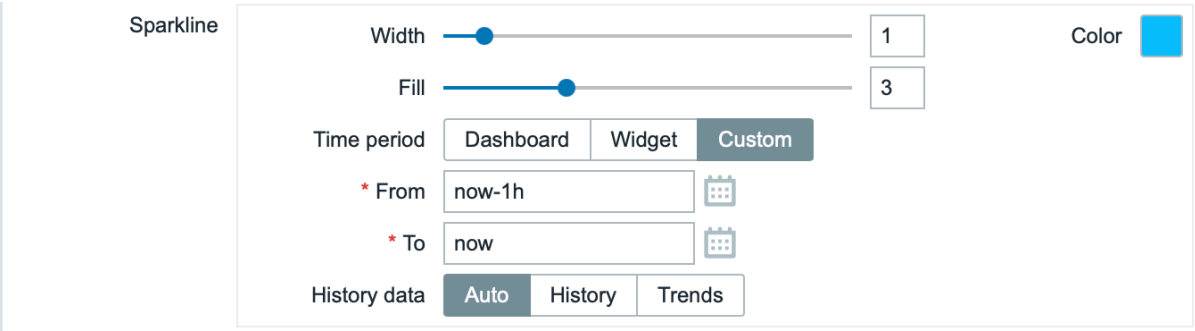
- Wskaźniki wzrostu (↑) i spadku (↓) nie są wyświetlane, jeśli dostępna jest tylko jedna wartość.
- Wartości są zawsze porównywane w tym samym zakresie; na przykład:
  - Najnowsza wartość jest porównywana z poprzednią wartością.
  - Wartość miesięczna jest porównywana z poprzednim miesiącem.
  - W przypadku agregacji poprzedni okres ma taki sam czas trwania jak wybrany okres i kończy się bezpośrednio przed rozpoczęciem wybranego okresu.

D oznacza kolor domyślny (zależny od motywu frontend). Aby przywrócić wartość domyślną, kliknij przycisk *Użyj domyślnego* w próbniku kolorów.

Wysokość wskaźnika zmiany jest równa rozmiarowi wartości (części całkowitej wartości w przypadku pozycji numerycznych).

Sparkline

Te parametry określają sposób wyświetlania wykresu sparkline.

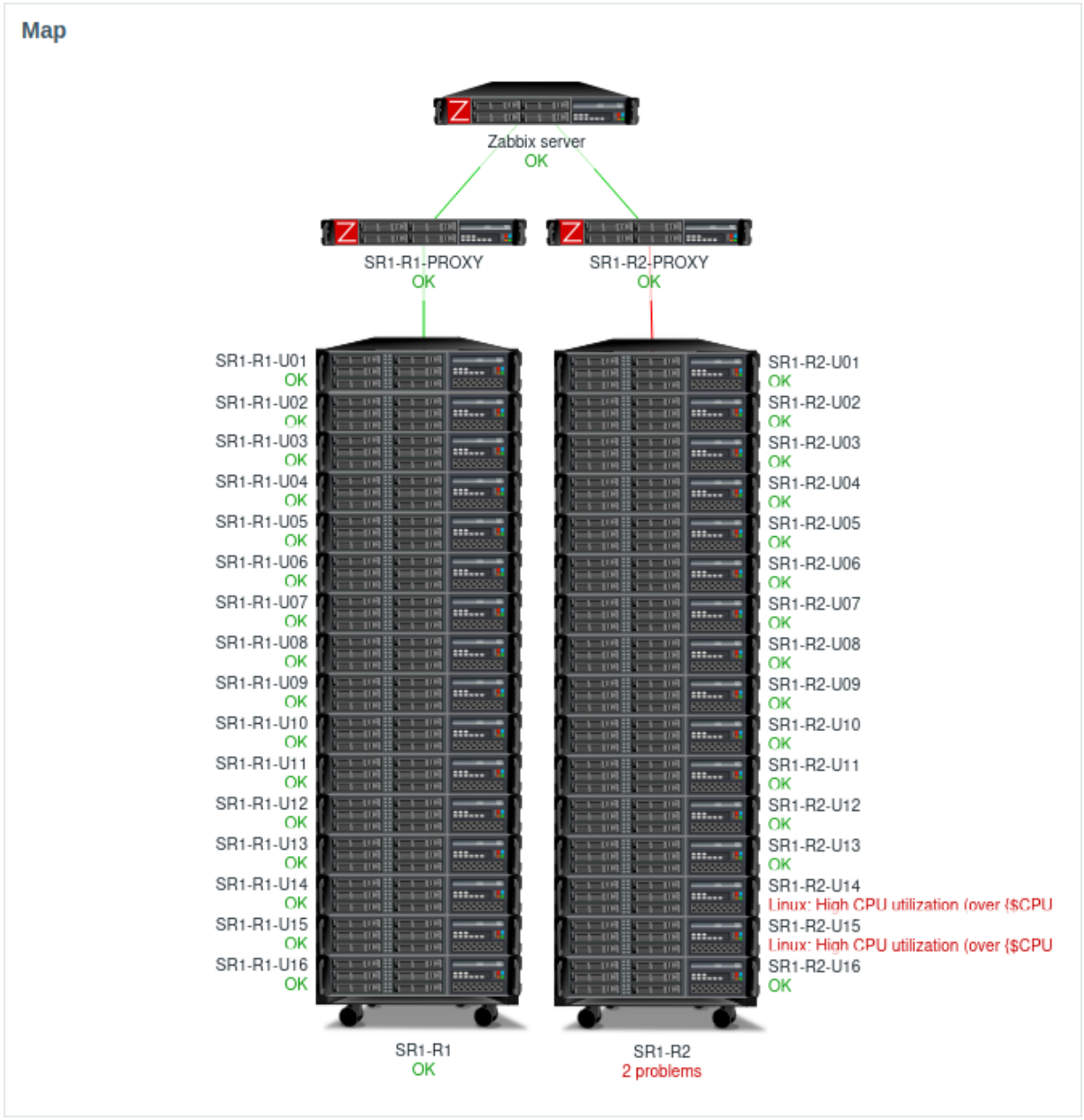


Width	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10.
Color	Wybierz kolor linii i wypełnienia.
Fill	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10.
Time period	Określ przedział czasu dla wartości, które mają zostać uwzględnione na wykresie sparkline. Wybierz źródło danych dla przedziału czasu: **Dashboard** - użyj selektora przedziału czasu pulpitu; **Widget** - użyj kompatybilnego widżetu (ustawionego w parametrze *Widget*); **Custom** - użyj niestandardowego przedziału czasu ustawionego w parametrach *From* i *To*; jeśli jest ustawiony, w prawym górnym rogu widżetu zostanie wyświetlona ikona zegara, wskazująca ustawiony czas po najechaniu kursorem. Zwróć uwagę, że niezależnie od konfiguracji *Time period* widżetu, kompatybilne widżety nadal mogą używać go jako źródła danych dla przedziału czasu. Ten parametr nie jest dostępny, jeśli *Aggregation function* jest ustawiona na „not used”.
Widget	Wprowadź lub wybierz kompatybilny widżet jako źródło danych dla przedziału czasu. Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na „Widget”.
From	Wprowadź lub wybierz początek przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na „Custom”.
To	Wprowadź lub wybierz koniec przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na „Custom”.
History data	Pobieraj dane z historii lub trendów: **Auto** - wybór automatyczny; **History** - pobieraj dane historyczne; **Trends** - pobieraj dane trendów.

Mapa

Przegląd

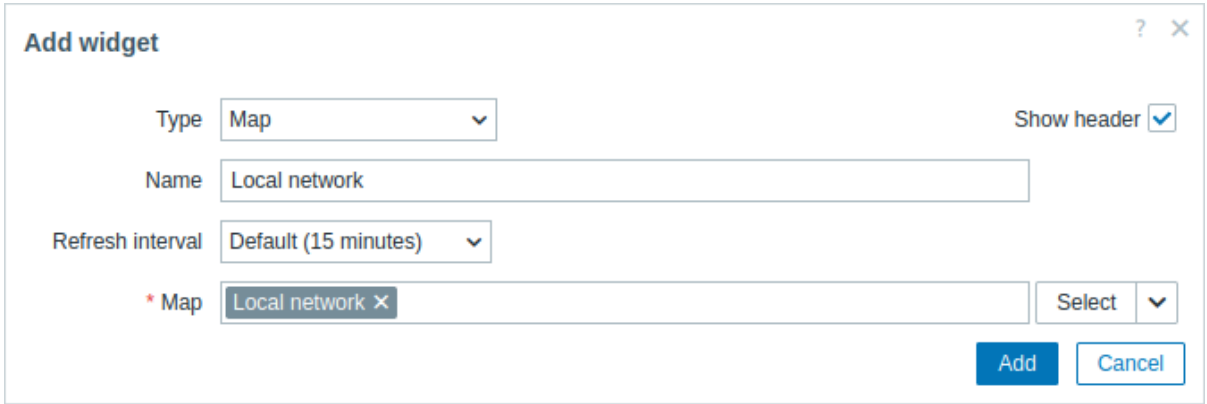
Widżet *Mapa* może wyświetlać mapę sieci lub podobną wizualizację, zapewniając dynamiczny przegląd sieci.



Można go połączyć z widżetem Drzewo nawigacji map, aby dynamicznie wyświetlać mapę wybraną w drzewie.

Konfiguracja

Aby skonfigurować, wybierz *Mapa* jako typ:



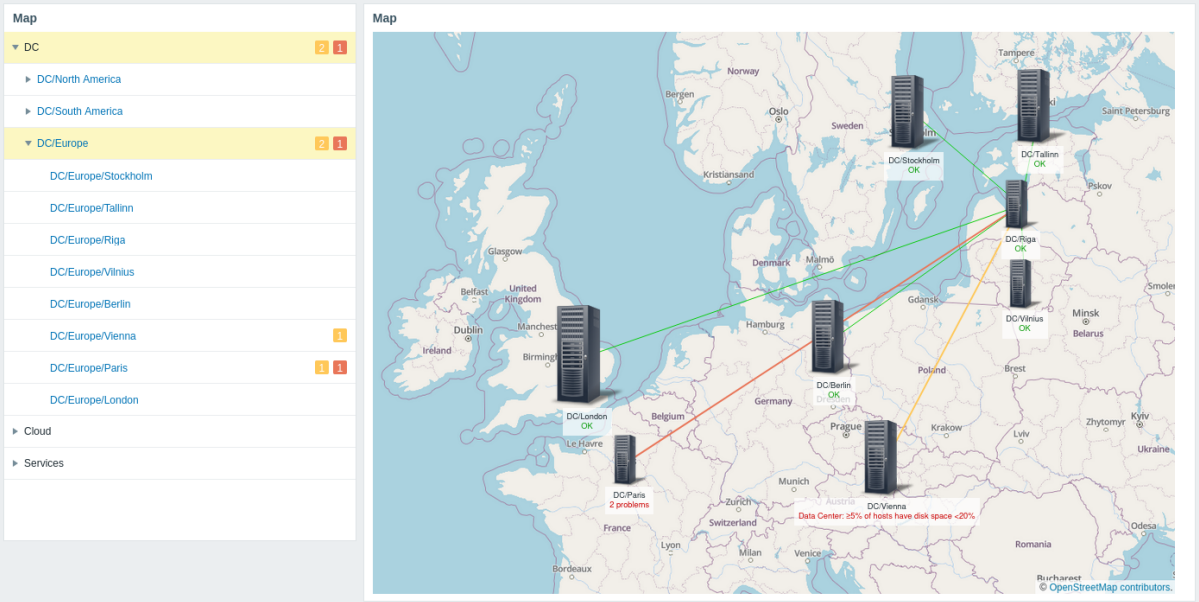
Oprócz parametrów wspólnych dla wszystkich widgetów możesz ustawić następujące opcje specyficzne:

Mapa	Ustaw mapę do wyświetlenia. Alternatywnie wybierz zgodny widget jako źródło danych dla wyświetlanej mapy. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy mapy lub widgetu zostanie wyświetlona rozwijana lista pasujących map lub widgetów.

Drzewo nawigacji mapy

Przegląd

Widżet *Drzewo nawigacji map* wyświetla hierarchię istniejących map oraz liczbę problemów dla każdej mapy i grupy map.

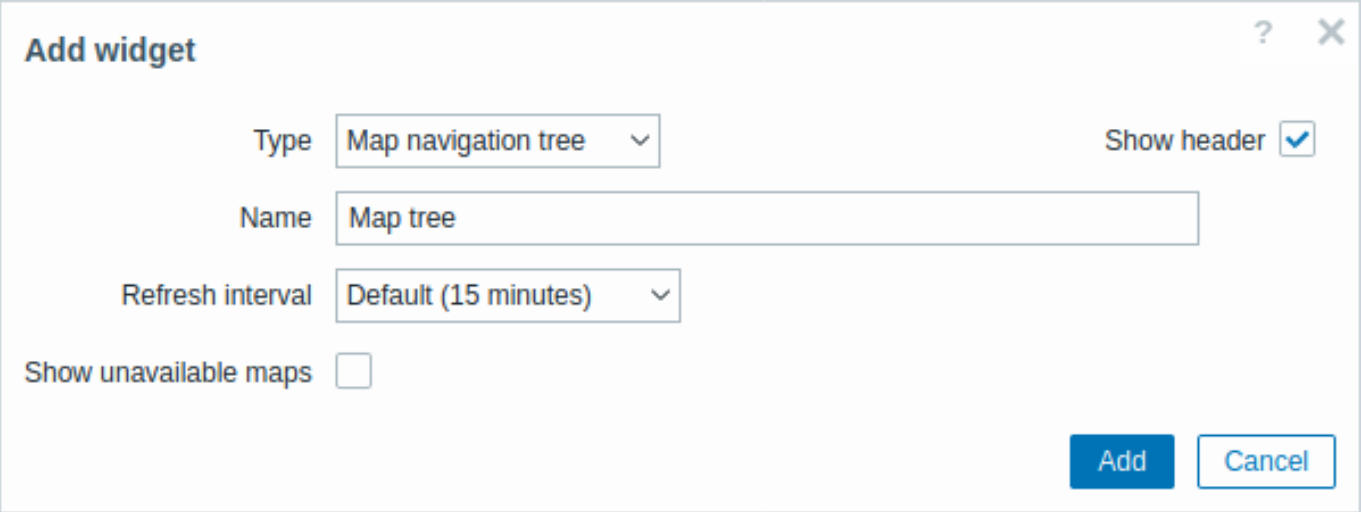


Można go połączyć z widżetem Mapa, aby dynamicznie wyświetlać mapę wybraną w drzewie.

Liczba problemów dla mapy najwyższego poziomu w hierarchii obejmuje wszystkie problemy podmap oraz problemy tej mapy.

Konfiguracja

Aby skonfigurować widżet, wybierz *Map navigation tree* jako typ:

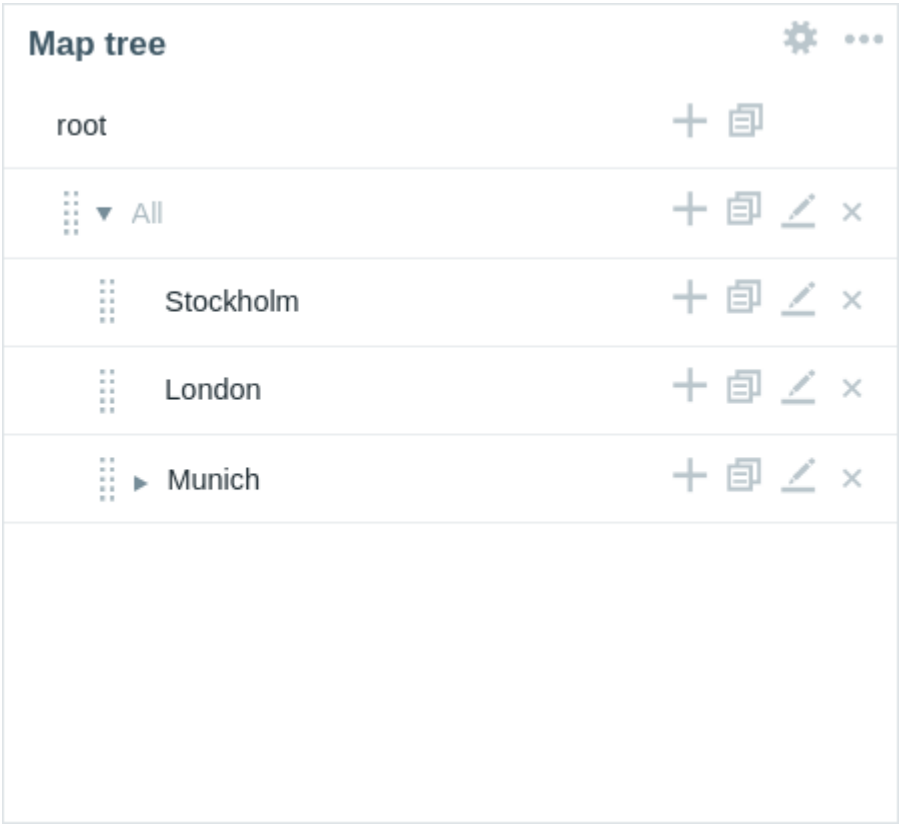


Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Pokaż niedostępne mapy	Zaznacz to pole wyboru, aby wyświetlać mapy, do których użytkownik nie ma uprawnień odczytu. Niedostępne mapy w drzewie nawigacji będą wyświetlane z wyszarzoną ikoną. Zwróć uwagę, że jeśli to pole wyboru jest zaznaczone, dostępne submaps są wyświetlane nawet wtedy, gdy mapa poziomu nadrzędnego jest niedostępna. Jeśli pole nie jest zaznaczone, dostępne submaps niedostępnej mapy nadrzędnej nie będą wyświetlane w ogóle. Liczba problemów jest obliczana na podstawie dostępnych map i dostępnych elementów mapy.

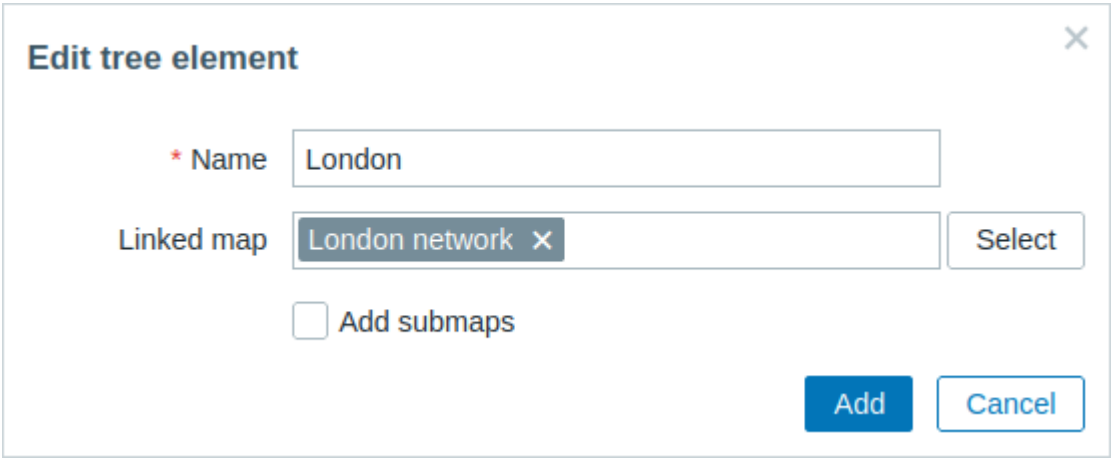
Elementy drzewa nawigacji są wyświetlane na liście. Możesz:

- przeciągnąć element (wraz z jego elementami podrzędnymi) w nowe miejsce na liście;
- rozwinąć lub zwinąć element, aby wyświetlić lub ukryć jego elementy podrzędne;
- dodać element podrzędny (z podlinkowaną mapą lub bez niej) do elementu;
- dodać wiele elementów podrzędnych (z podlinkowanymi mapami) do elementu;
- edytować element;
- usunąć element (wraz z jego elementami podrzędnymi).



Konfiguracja elementu

Aby skonfigurować element drzewa nawigacji, dodaj nowy element lub edytuj istniejący element.



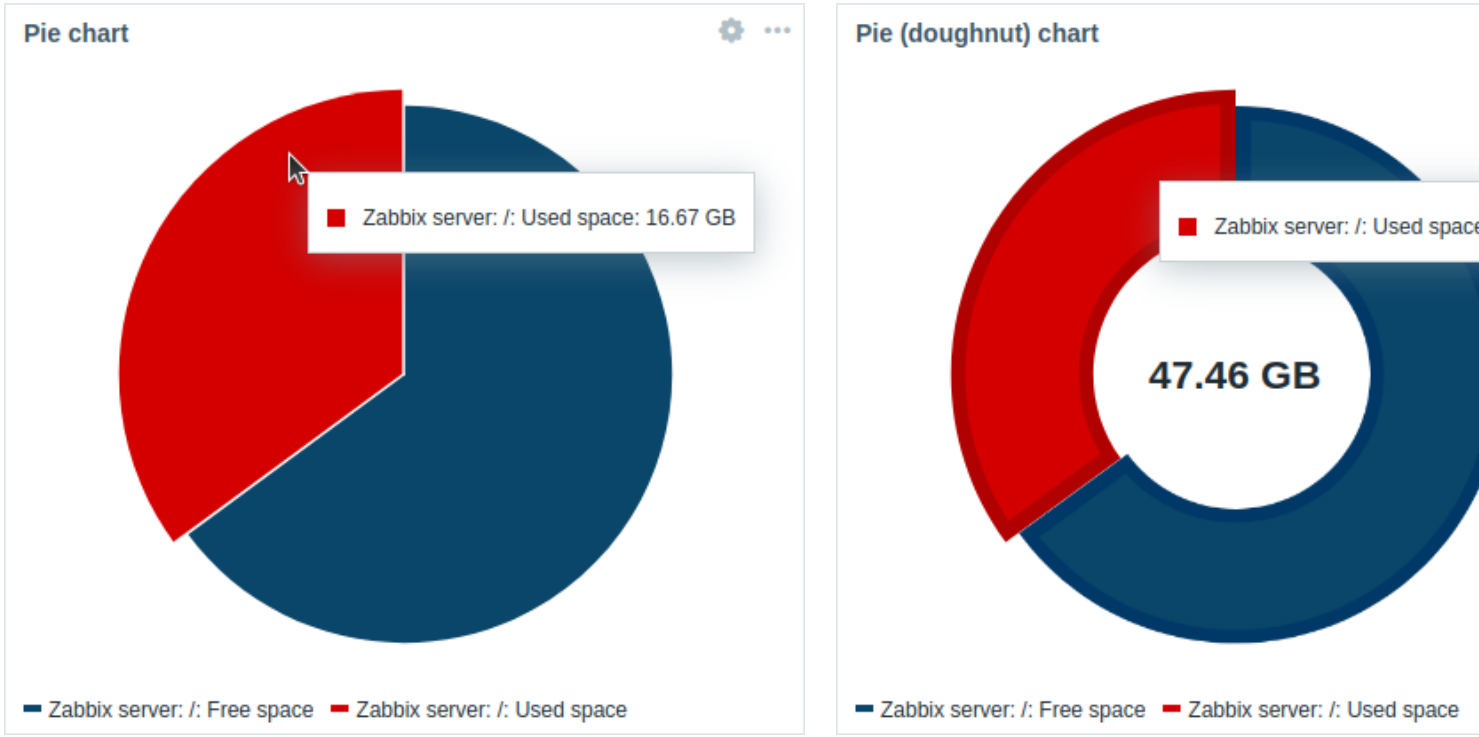
Dostępne są następujące parametry konfiguracji elementu drzewa nawigacji:

Nazwa	Wprowadź nazwę elementu drzewa nawigacji.
Powiązana mapa	Wybierz mapę, którą chcesz powiązać z elementem drzewa nawigacji. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy mapy zostanie wyświetlona rozwijana lista pasujących map.
Dodaj podmapy	Zaznacz to pole wyboru, aby dodać podmapy powiązanej mapy jako elementy podrzędne elementu drzewa nawigacji.

Wykres kołowy

Przegląd

Widżet *Wykres kołowy* wyświetla dane liczbowe z pozycji w postaci opartego na wektorach wykresu kołowego lub pierścieniowego. Widżet ten może pomóc w wizualizacji udziału pozycji lub hostów w całym zbiorze danych.

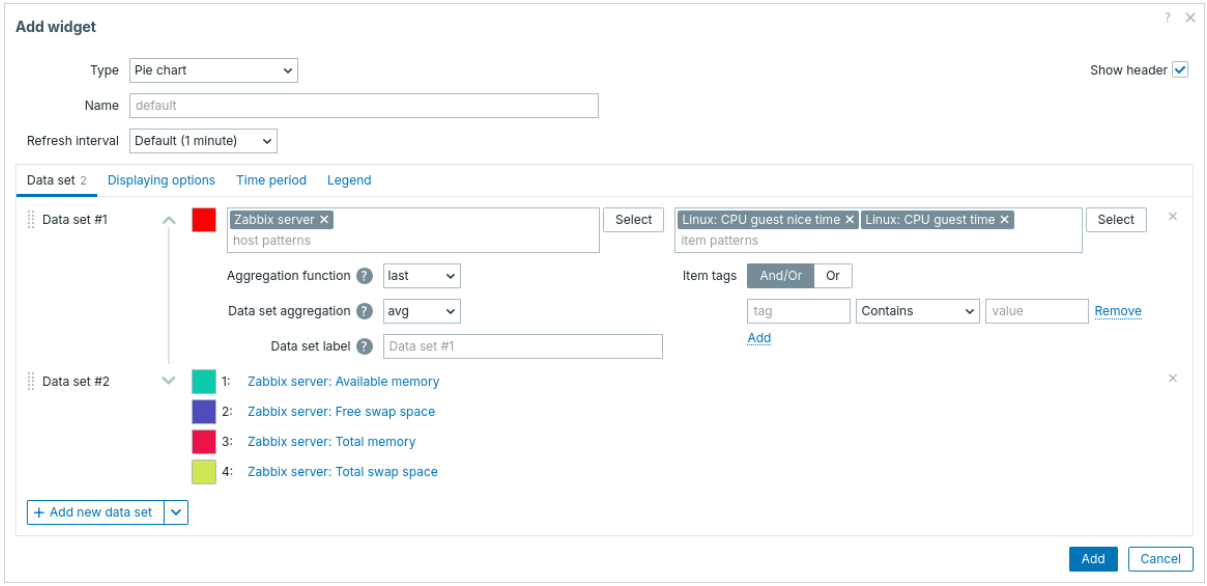


Po najechaniu kursorem sektor zostaje powiększony i wyświetla swoją legendę; kliknięcie sektora powoduje pozostawienie go w powiększeniu, a jego legenda pozostaje widoczna do momentu zamknięcia.

Informacje wyświetlane w widżecie *Wykres kołowy* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu *Działania* widżetu ⋯, które pojawia się po najechaniu kursorem na widżet. W zależności od ustawień przeglądarki obraz zostanie automatycznie zapisany w folderze `Downloads` albo otworzy się okno dialogowe *Zapisz*, umożliwiające wybranie lokalizacji.

Konfiguracja

Aby skonfigurować, wybierz *Pie chart* jako typ:



Zestaw danych

Karta **Zestaw danych** umożliwia wybranie danych do wykresu kołowego przez dodanie zestawów danych. Można dodać dwa typy zestawów danych:

- *Wzorce pozycji* - wyświetlane są dane z pasujących pozycji. Można wybrać pojedynczy kolor bazowy lub wiersz palety, aby przypisać różne kolory do każdej pasującej pozycji.
- *Lista pozycji* - wyświetlane są dane z wybranych pozycji. Kolor każdej pozycji można wybrać indywidualnie za pomocą selektora.

Domyślnie dodawany jest zestaw danych typu *Wzorce pozycji*.

Zestaw danych	Dla zestawu danych **Wzorce pozycji**: Wybierz lub wprowadź wzorce hostów i pozycji; dane pozycji pasujących do wprowadzonych wzorców będą wyświetlane na wykresie kołowym; można wyświetlić maksymalnie 50 pozycji. Do wyboru można używać wzorców wieloznacznych (na przykład ∗ zwróci wyniki pasujące do zera lub większej liczby znaków). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij klawisz *Enter*. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać indywidualnie na przykład pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (na przykład *item2*, *item3*). Określenie wzorców hostów i pozycji jest wymagane dla zestawów danych typu ”Wzorce pozycji”. Zobacz także: Szczegóły konfiguracji zestawu danych. Dla zestawu danych **Lista pozycji**: Wybierz pozycje dla wykresu kołowego, klikając przycisk *Dodaj pozycję*. Lista rozwijana *Typ* znajdująca się za nazwą pozycji umożliwia wybranie typu wyświetlania dla każdej pozycji: **Normalny** - wartość pozycji jest przedstawiana proporcjonalnie na wykresie kołowym (domyślnie); **Całkowita** - wartość pozycji zajmuje cały wykres kołowy. Należy pamiętać, że na jednym wykresie kołowym może istnieć tylko jedna pozycja typu ”Całkowita” i zostanie ona umieszczona jako pierwsza w legendzie wykresu kołowego. Jeśli pozycja ma ustawiony typ ”Całkowita”, parametr *Agregacja zestawu danych* (opisany poniżej) zostanie wyłączony i ustawiony na ”nieużywany”. Można również wybrać kompatybilne widżety jako źródło danych dla pozycji, klikając przycisk *Dodaj widżet*. Do widżetów mają zastosowanie te same opcje co do pojedynczych pozycji. Określenie pozycji lub widżetów jest wymagane dla zestawów danych typu ”Lista pozycji”. Zobacz także: Szczegóły konfiguracji zestawu danych. Należy pamiętać, że dozwolone są tylko numeryczne typy pozycji. Podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu parametr określania wzorców hostów jest niedostępny, a parametr określania listy pozycji umożliwia wybranie tylko pozycji skonfigurowanych w szablonie.
Funkcja agregacji	Określ, której funkcji agregacji użyć dla każdej pozycji w zestawie danych w wybranym *Okresie*: **min** - wyświetla najmniejszą wartość; **max** - wyświetla największą wartość; **avg** - wyświetla wartość średnią; **sum** - wyświetla sumę wartości; **count** - wyświetla liczbę wartości; **first** - wyświetla pierwszą wartość; **last** - wyświetla ostatnią wartość (domyślnie). Agregacja umożliwia wyświetlenie zagregowanej wartości dla określonego przedziału (5 minut, godziny, dnia). Dla funkcji *min*, *max*, *avg* i *sum* można wyświetlać tylko dane numeryczne. W przypadku funkcji *count* dane nienumeryczne zostaną zmienione na numeryczne.
Agregacja zestawu danych	Określ, której funkcji agregacji użyć dla całego zestawu danych: **nieużywany** - brak agregacji, pozycje są wyświetlane osobno (domyślnie); **min** - wyświetla najmniejszą wartość; **max** - wyświetla największą wartość; **avg** - wyświetla wartość średnią; **sum** - wyświetla sumę wartości; **count** - wyświetla liczbę wartości. Agregacja umożliwia wyświetlenie zagregowanej wartości dla określonego przedziału (5 minut, godziny, dnia) wybranego na karcie *Okres* lub używanego dla całego pulpitu nawigacyjnego.

Etykieta zestawu danych	Określ niestandardową etykietę zestawu danych. Etykieta jest wyświetlana w konfiguracji zestawu danych oraz w legendzie wykresu kołowego (dla zagregowanych zestawów danych). Wszystkie zestawy danych są numerowane, w tym te, dla których określono *Etykietę zestawu danych*. Jeśli nie określono etykiety, zestaw danych zostanie oznaczony automatycznie zgodnie z numerem (np. ”Zestaw danych nr 2”, ”Zestaw danych nr 3” itd.). Numeracja zestawów danych jest przeliczana po zmianie kolejności lub przeciągnięciu zestawów danych. Etykiety zestawów danych, które są zbyt długie, zostaną skrócone, aby zmieścić się w miejscu wyświetlania (np. ”Liczba proc...”).
Tagi pozycji	Określ tagi używane do filtrowania pozycji wyświetlanych w widżecie. Można uwzględniać lub wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnia określone nazwy tagów; **Równa się** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucza określone nazwy tagów; **Nie równa się** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Or; **Or** - wystarczy spełnienie jednego warunku.

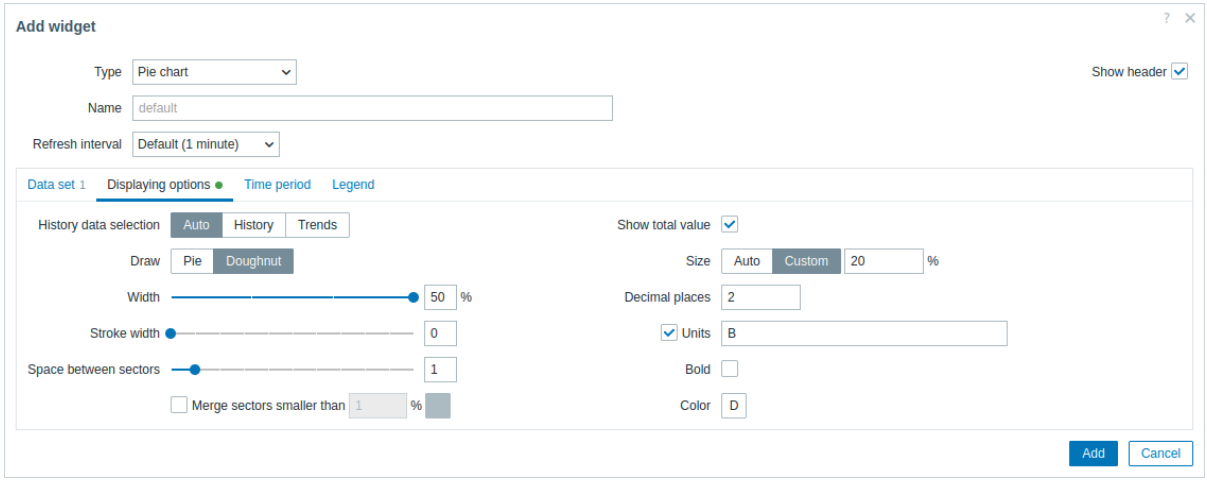
Szczegóły konfiguracji zestawu danych

Istniejące zestawy danych są wyświetlane na liście. Możesz zmieniać ich kolejność, rozwijać/zwijać, zmieniać kolory oraz klonować te zestawy danych.

Więcej informacji można znaleźć w sekcji dotyczącej szczegółów konfiguracji zestawu danych widżetu *Graph*. Te szczegóły mają również zastosowanie do widżetu *Pie chart*.

Opcje wyświetlania

Karta **Opcje wyświetlania** umożliwia zdefiniowanie opcji wyboru danych historycznych i wizualizacji dla wykresu kołowego:

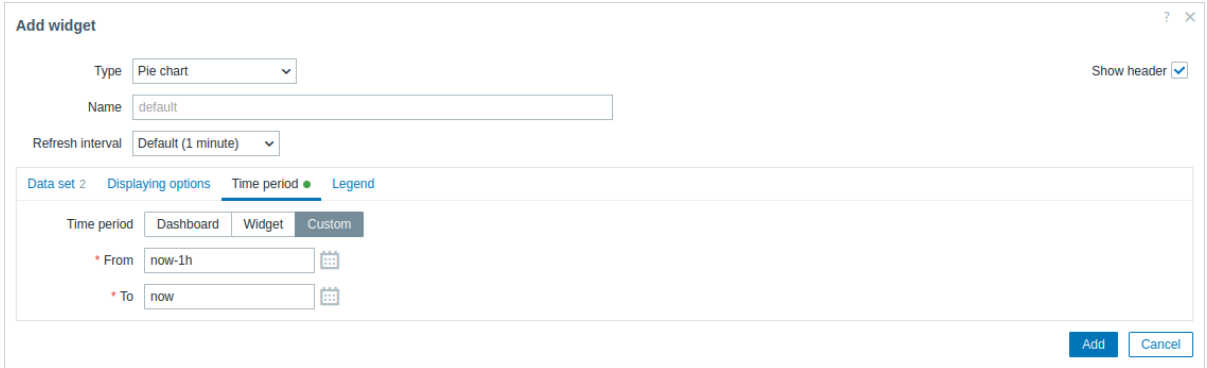


Wybór danych historycznych	Wybierz źródło danych: **Auto** - dane są pobierane zgodnie z klasycznym algorytmem (domyślnie); **History** - dane z historii; **Trends** - dane z trendów.

Rysowanie	Wybierz styl wizualizacji wykresu kołowego: **Pie** - pełne koło (sektory zajmują 100% promienia); **Doughnut** - koło z pustym środkiem (sektory zajmują do 50% promienia).
Odstęp między sek-torami	Wybierz rozmiar odstępu (w jednostkach od 0 do 10) między sektorami (domyślnie „1”).
Scal sek-tory mniejsze niż N%	Zaznacz pole wyboru, aby scalić sektory mniejsze niż N%. Jeśli opcja jest włączona, wybierz kolor scalonych sektorów oraz próg procentowy (N) scalania małych sektorów.
Styl rysowa-nia: ***Dough-nut***	
Szerokość	Wybierz szerokość pierścienia: 20, 30, 40 lub 50% (domyślnie) promienia.
Szerokość obramowa-nia	Wybierz szerokość obramowania sektora pierścienia (0-10).
Pokaż wartość całkow-itą	Zaznacz pole wyboru, aby wyświetlić wartość całkowitą na środku wykresu pierścieniowego.
Rozmiar	Wybierz opcję rozmiaru wartości całkowitej: **Auto** - rozmiar tekstu jest automatycznie dopasowywany, aby był czytelny i mieścił się w środku pierścienia; **Custom** - określ rozmiar tekstu jako procent wysokości całego widgetu.
Miejsca dziesiętne	Określ liczbę miejsc dziesiętnych dla wartości całkowitej (0-6).
Jednostki	Określ jednostki dla wartości całkowitej.
Pogrubienie	Zaznacz pole wyboru, aby wyświetlać wartość całkowitą pogrubioną czcionką.
Kolor	Wybierz kolor wartości całkowitej.

Okres

Karta **Okres** umożliwia ustawienie niestandardowego okresu dla ustawień agregacji wykresu kołowego:

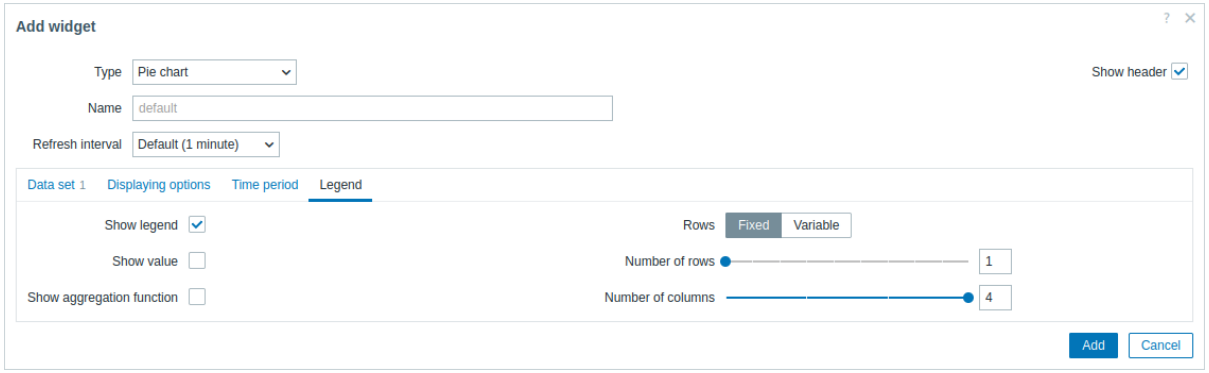


Okres	Wybierz źródło danych dla okresu: **Dashboard** - użyj selektora okresu dashboardu; **Widget** - użyj zgodnego widgetu (ustawionego w parametrze *Widget*); **Niestandardowy** - użyj niestandardowego okresu określonego w parametrach *Od* i *Do*; jeśli zostanie ustawiony, w prawym górnym rogu widgetu będzie wyświetlana ikona zegara wskazująca ustawiony czas po najechaniu kursorem myszy. Należy pamiętać, że niezależnie od konfiguracji *Okresu* widgetu zgodne widgety nadal mogą używać go jako źródła danych dla okresu.
Widget	Wprowadź lub wybierz zgodny widget (*Graph*, *Graph (classic)*, *Graph prototype*) jako źródło danych dla okresu. Ten parametr jest dostępny, jeśli dla parametru *Okres* ustawiono wartość „Widget”.

Od	Wprowadź lub wybierz początek okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla parametru *Okres* ustawiono wartość „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla parametru *Okres* ustawiono wartość „Niestandardowy”.

Legenda

Karta **Legenda** umożliwia dostosowanie legendy wykresu kołowego:

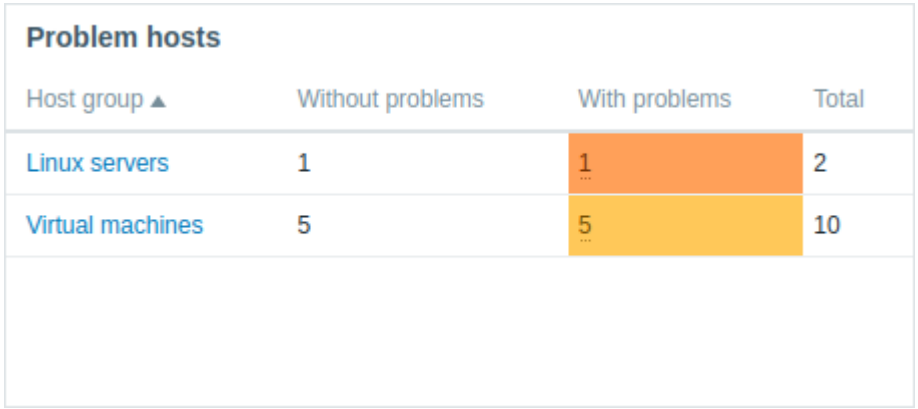


Pokaż legendę	Odznacz to pole wyboru, aby ukryć legendę na wykresie kołowym (domyślnie zaznaczone).
Pokaż wartość	Zaznacz to pole wyboru, aby wyświetlić wartość pozycji w legendzie.
Pokaż funkcję agregacji	Zaznacz to pole wyboru, aby wyświetlić funkcję agregacji w legendzie.
Wiersze	Wybierz tryb wyświetlania wierszy legendy: **Stała** - liczba wyświetlanych wierszy jest określana przez wartość parametru *Liczba wierszy*; **Zmienna** - liczba wyświetlanych wierszy jest określana przez liczbę skonfigurowanych pozycji, ale nie może przekroczyć wartości parametru *Maksymalna liczba wierszy*.
Liczba wierszy/ Maksymalna liczba wierszy	Jeśli dla opcji *Wiersze* ustawiono wartość „Stała”, ustaw liczbę wyświetlanych wierszy legendy (1-10). Jeśli dla opcji *Wiersze* ustawiono wartość „Zmienna”, ustaw maksymalną liczbę wyświetlanych wierszy legendy (1-10).
Liczba kolumn	Ustaw liczbę wyświetlanych kolumn legendy (1-4). Ten parametr jest dostępny, jeśli opcja *Pokaż wartość* jest odznaczona.

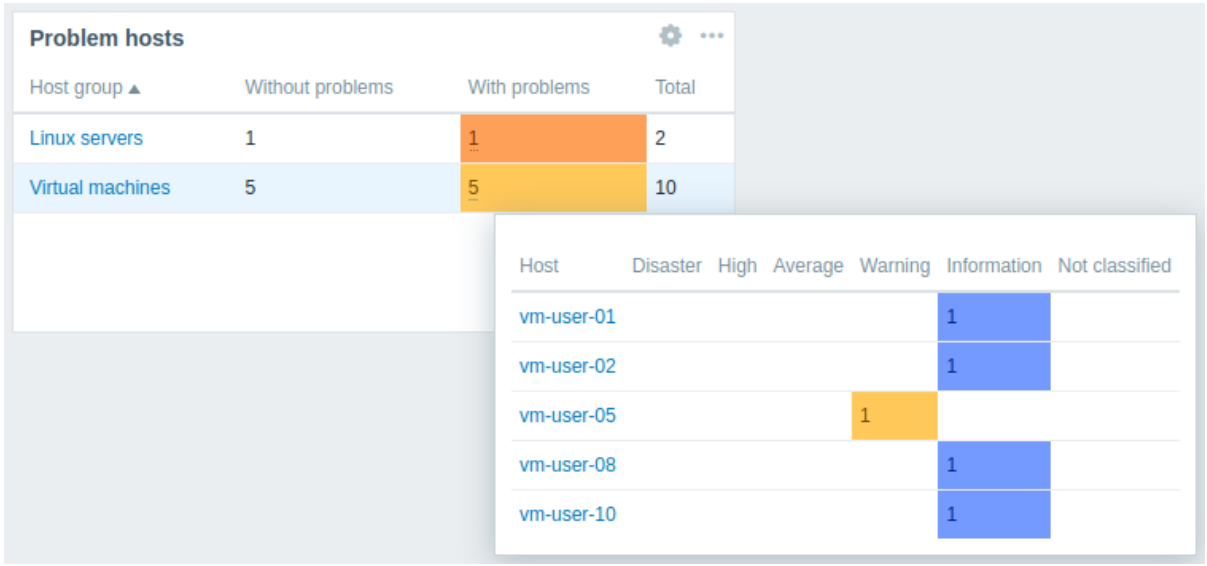
Hosty z problemami

Przegląd

Widżet *Hosty z problemami* wyświetla liczbę problemów przyczynowych dla każdej grupy hostów oraz pokazuje najwyższy poziom ważności problemu w każdej grupie. Umożliwia filtrowanie problemów według różnych parametrów (grupy hostów, hosty, nazwy problemów itp.).

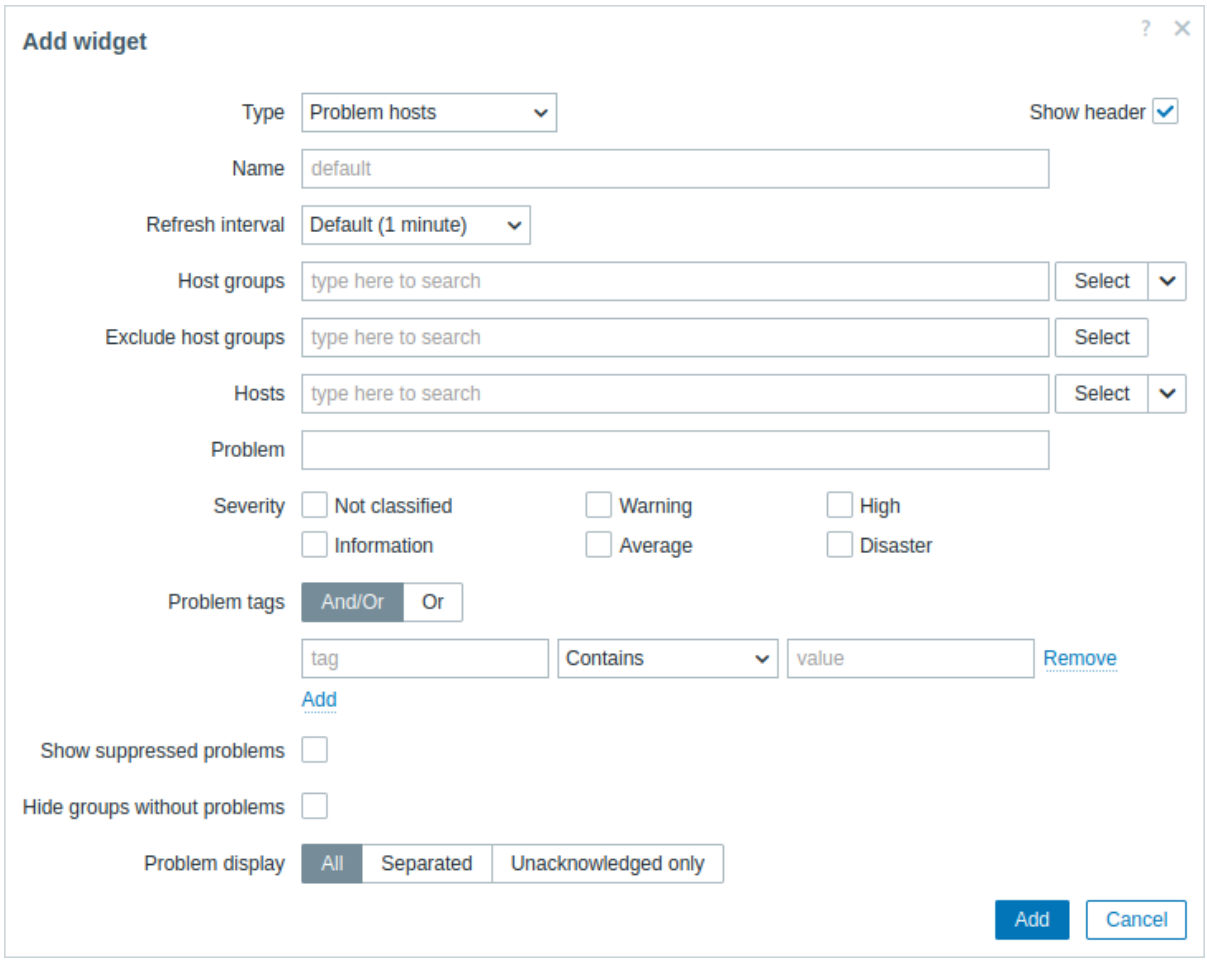


Po najechaniu kursorem na liczbę problemów wyświetlana jest lista hostów z problemami:



Konfiguracja

Aby skonfigurować, wybierz *Problem hosts* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów, możesz ustawić następujące opcje specyficzne:

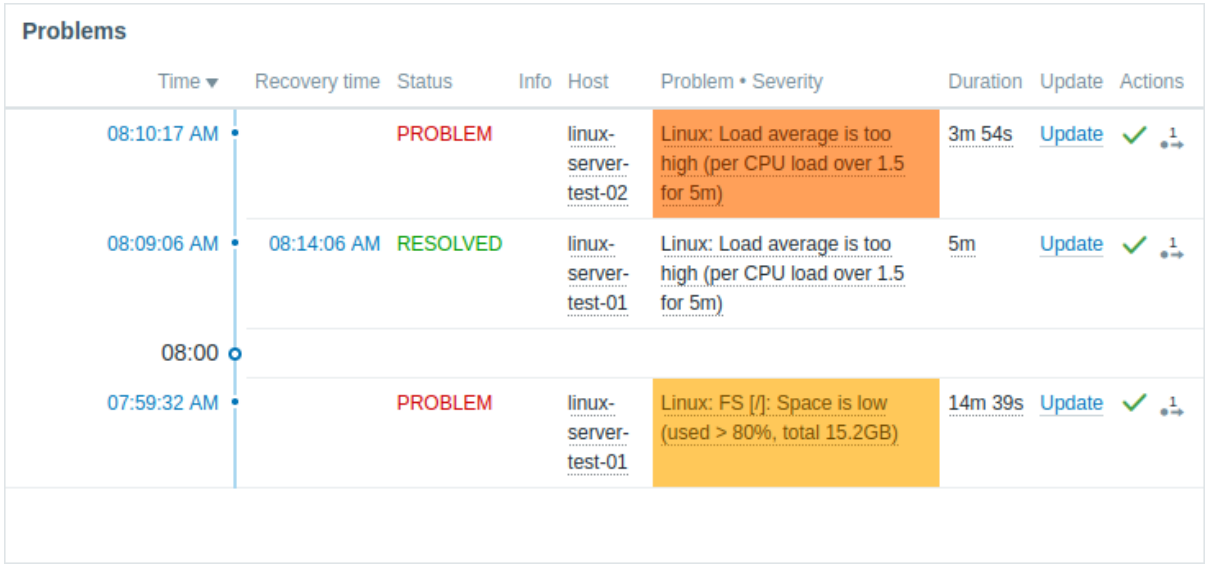
Host groups	Wybierz grupy hostów, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną wprowadzone żadne grupy hostów, wyświetlone zostaną wszystkie grupy hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

Exclude host groups	Wybierz grupy hostów, które mają zostać ukryte w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów nie będą wyświetlane w widżecie. Na przykład hosty 001, 002, 003 mogą należeć do Grupy A, a hosty 002, 003 również do Grupy B. Jeśli jednocześnie wybierzemy *show* dla Grupy A i *exclude* dla Grupy B, na pulpicie zostaną wyświetlone tylko dane hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Hosts	Wybierz hosty, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli nie zostaną wprowadzone żadne hosty, wyświetlone zostaną wszystkie hosty. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Problem	Możesz ograniczyć liczbę hostów z problemami wyświetlanych według nazwy problemu. Jeśli wpiszesz tutaj ciąg znaków, wyświetlone zostaną tylko te hosty, które mają problemy, których nazwa zawiera podany ciąg. Makra nie są rozwijane.
Severity	Zaznacz poziomy ważności problemów, aby filtrować problemy wyświetlane w widżecie. Jeśli nie zostanie zaznaczony żaden poziom ważności, wyświetlone zostaną wszystkie problemy.
Problem tags	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane przez warunek *Or*; **Or** - wystarczy spełnienie jednego warunku.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby tłumione (niewyświetlane) z powodu konserwacji hosta.
Hide groups without problems	Zaznacz opcję *Hide groups without problems*, aby ukryć w widżecie dane z grup hostów bez problemów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Problem display	Wyświetl liczbę problemów jako: **All** - zostanie wyświetlona pełna liczba problemów; **Separated** - liczba niepotwierdzonych problemów zostanie wyświetlona oddzielnie jako część całkowitej liczby problemów; **Unacknowledged only** - zostanie wyświetlona tylko liczba niepotwierdzonych problemów.

Problemy

Przegląd

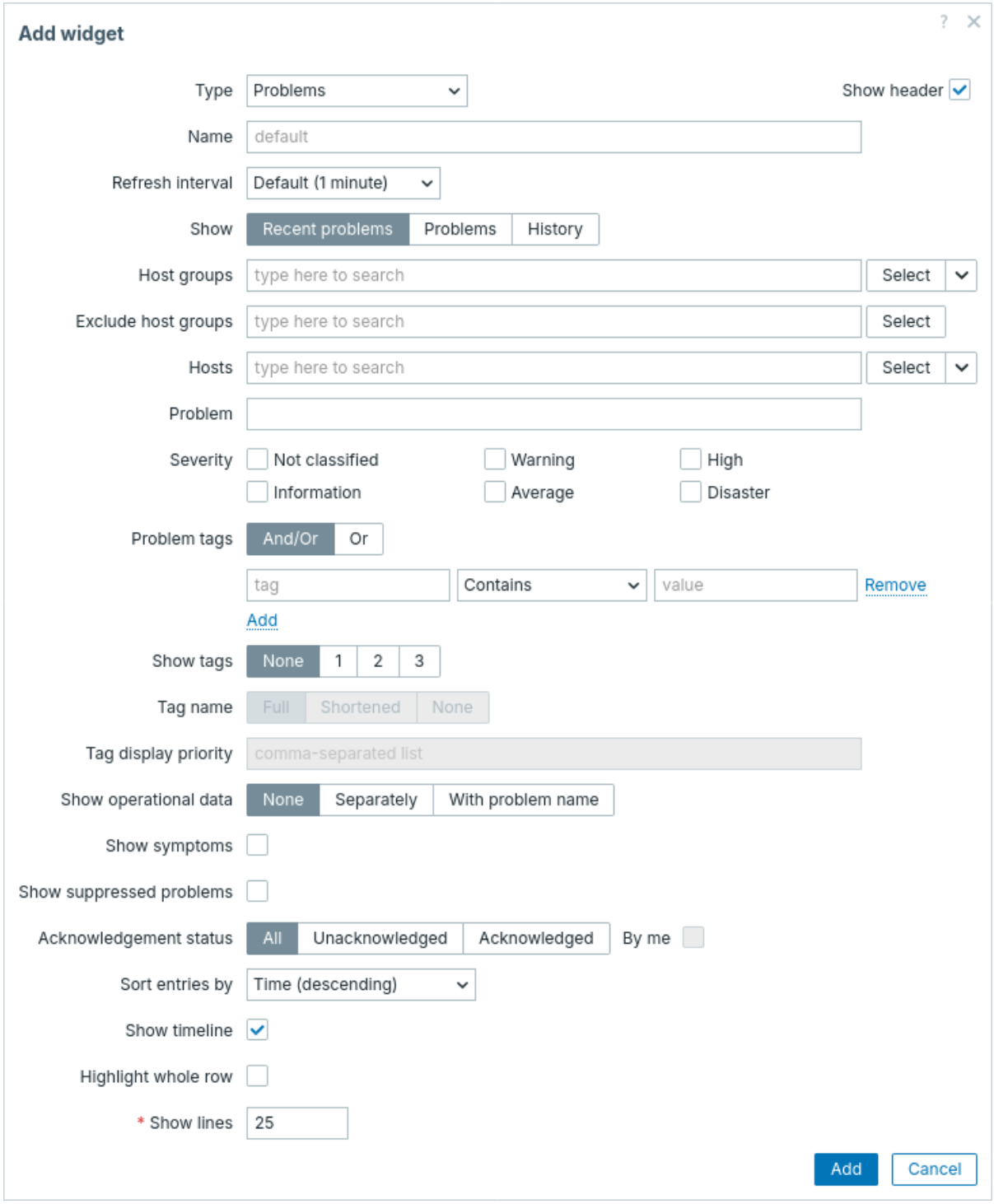
Widżet *Problemy* wyświetla bieżące problemy filtrowane według różnych parametrów (grupy hostów, hosty, nazwy problemów itp.), zapewniając przejrzysty widok tego, co wymaga uwagi.



Wyświetla te same dane co *Monitoring* > *Problemy* i może wyświetlić do 1000 rekordów.

Konfiguracja

Aby skonfigurować, wybierz *Problems* jako typ:



Możesz ograniczyć liczbę problemów wyświetlanych w widżecie na różne sposoby - według statusu problemu, nazwy problemu, ważności, grupy hostów, hosta, tagu zdarzenia, statusu potwierdzenia itd.

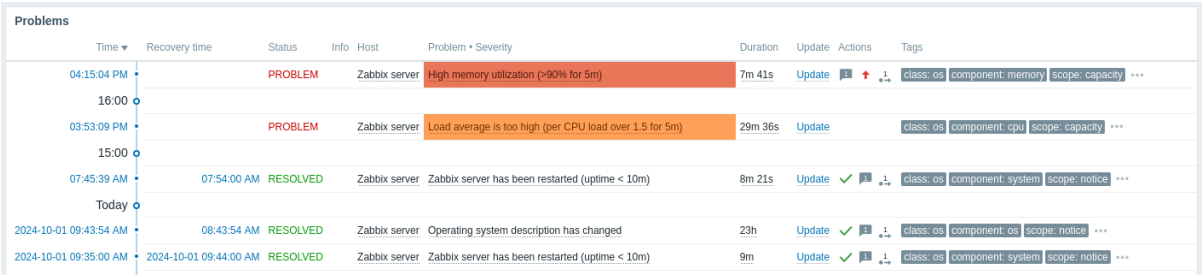
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

Show	Filtruj według statusu problemu: **Recent problems** - wyświetlane są nierozwiązane i niedawno rozwiązane problemy (domyślnie); **Problems** - wyświetlane są nierozwiązane problemy; **History** - wyświetlana jest historia wszystkich zdarzeń.

Host groups	Wybierz grupy hostów, których problemy mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Problemy z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną podane żadne grupy hostów, będą wyświetlane problemy ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Exclude host groups	Wybierz grupy hostów, których problemy mają być ukrywane w widżecie. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Problemy z tych grup hostów nie będą wyświetlane w widżecie. Na przykład hosty 001, 002, 003 mogą należeć do Group A, a hosty 002, 003 także do Group B. Jeśli wybierzemy jednocześnie *show* Group A i *exclude* Group B, w widżecie będą wyświetlane tylko problemy z hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Hosts	Wybierz hosty, których problemy mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Jeśli nie zostaną podane żadne hosty, będą wyświetlane problemy ze wszystkich hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na dashboardzie szablonu.
Problem	Możesz ograniczyć liczbę wyświetlanych problemów według ich nazwy. Jeśli wpiszesz tutaj ciąg znaków, wyświetlone zostaną tylko te problemy, których nazwa zawiera wpisany ciąg. Makra nie są rozwijane.
Severity	Zaznacz ważności problemów, aby filtrować problemy wyświetlane w widżecie. Jeśli nie zostanie zaznaczona żadna ważność, wyświetlane będą wszystkie problemy.
Problem tags	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wpisany ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wpisany ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku *Or*; **Or** - wystarczy spełnienie jednego warunku. Po zastosowaniu filtra tagi określone tutaj będą wyświetlane jako pierwsze wraz z problemem, chyba że zostanie to zastąpione przez listę *Tag display priority* (patrz poniżej).
Show tags	Wybierz liczbę wyświetlanych tagów: **None** - brak kolumny *Tags*; **1** - kolumna *Tags* zawiera jeden tag; **2** - kolumna *Tags* zawiera dwa tagi; **3** - kolumna *Tags* zawiera trzy tagi. Aby zobaczyć wszystkie tagi problemu, najedź kursorem myszy na ikonę trzech kropek.
Tag name	Wybierz tryb wyświetlania nazw tagów: **Full** - nazwy i wartości tagów są wyświetlane w pełnej postaci; **Shortened** - nazwy tagów są skrócone do 3 symboli, ale wartości tagów są wyświetlane w pełnej postaci; **None** - wyświetlane są tylko wartości tagów; bez nazw.

Tag display priority	Wprowadź priorytet wyświetlania tagów dla problemu jako listę tagów oddzielonych przecinkami. Należy używać tylko nazw tagów, bez wartości. Przykład: `Services,Applications,Application` Tagi z tej listy będą zawsze wyświetlane jako pierwsze, zastępując naturalną kolejność alfabetyczną.
Show operational data	Wybierz tryb wyświetlania danych operacyjnych: **None** - dane operacyjne nie są wyświetlane; **Separately** - dane operacyjne są wyświetlane w osobnej kolumnie; **With problem name** - dane operacyjne są dołączane do nazwy problemu, z użyciem nawiasów dla danych operacyjnych. Szczegóły znajdziesz w opisie pola *Operational data* w konfiguracji wyzwalacza.
Show symptoms	Zaznacz pole wyboru, aby wyświetlać na osobnej linii problemy sklasyfikowane jako objawy.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby ukryte (niewyświetlane) z powodu konserwacji hosta lub pojedynczego ukrywania problemu.
Acknowledgement status	Filtruj, aby wyświetlać wszystkie problemy, tylko niepotwierdzone problemy lub tylko potwierdzone problemy. Zaznacz dodatkowe pole wyboru, aby odfiltrować problemy kiedykolwiek przez Ciebie potwierdzone.
Sort entries by	Sortuj wpisy według: **Time** (malejąco lub rosnąco); **Severity** (malejąco lub rosnąco); **Problem name** (malejąco lub rosnąco); **Host** (malejąco lub rosnąco). Sortowanie wpisów według **Host** (malejąco lub rosnąco) nie jest dostępne podczas konfigurowania widżetu na dashboardzie szablonu.
Show timeline	Zaznacz pole wyboru, aby wyświetlić wizualną oś czasu.
Highlight whole row	Zaznacz pole wyboru, aby podświetlać cały wiersz dla nierozwiązanych problemów. Do podświetlania używany jest kolor ważności problemu. *Highlight whole row* nie jest dostępne w motywach o wysokim kontraście.
Show lines	Określ liczbę wierszy problemów do wyświetlenia.

Korzystanie z widżetu

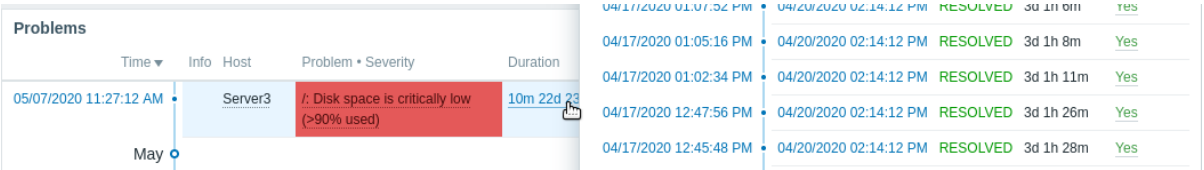


Widżet problemów zapewnia szybki dostęp do dodatkowych informacji:

- Kliknij datę problemu (w kolumnie Czas) lub datę odzyskania (w kolumnie Czas odzyskania), aby wyświetlić szczegóły zdarzenia.
- Jeśli pole Info nie jest puste, najedź kursorem na wyświetloną ikonę, aby zobaczyć dodatkowe szczegóły.
- Kliknij nazwę hosta, aby otworzyć menu hosta.
- Kliknij nazwę problemu, aby otworzyć menu zdarzenia.
- Najedź kursorem na czas trwania problemu lub kliknij go, aby wyświetlić wyskakujące okno zdarzenia problemu.
- Kliknięcie *Aktualizuj* otwiera okno Aktualizacja problemu.
- Najedź kursorem na szarą ikonę strzałki w kolumnie Akcje lub kliknij ją, aby wyświetlić listę wykonanych akcji.

Wyskakujące okno zdarzenia problemu

Wyskakujące okno zdarzenia problemu zawiera listę zdarzeń problemu dla tego wyzwalacza oraz, jeśli zostały zdefiniowane, opis wyzwalacza i klikalny adres URL.



Aby wyświetlić wyskakujące okno zdarzenia problemu:

- Najedź kursorem na czas trwania problemu w kolumnie *Czas trwania* widżetu *Problemy*. Wyskakujące okno znika po odsunięciu kursora od czasu trwania.
- Kliknij czas trwania w kolumnie *Czas trwania* widżetu *Problemy*. Wyskakujące okno znika tylko wtedy, gdy ponownie klikniesz czas trwania.

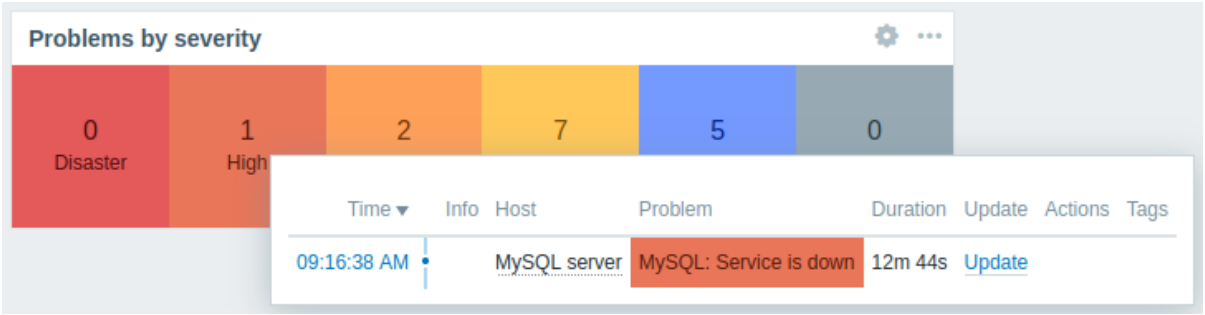
Problemy według poziomu ważności

Przegląd

Widżet *Problemy według ważności* wyświetla liczbę problemów przyczynowych pogrupowanych według ważności. Umożliwia filtrowanie problemów według różnych parametrów (grupy hostów, hosty, nazwy problemów itp.).

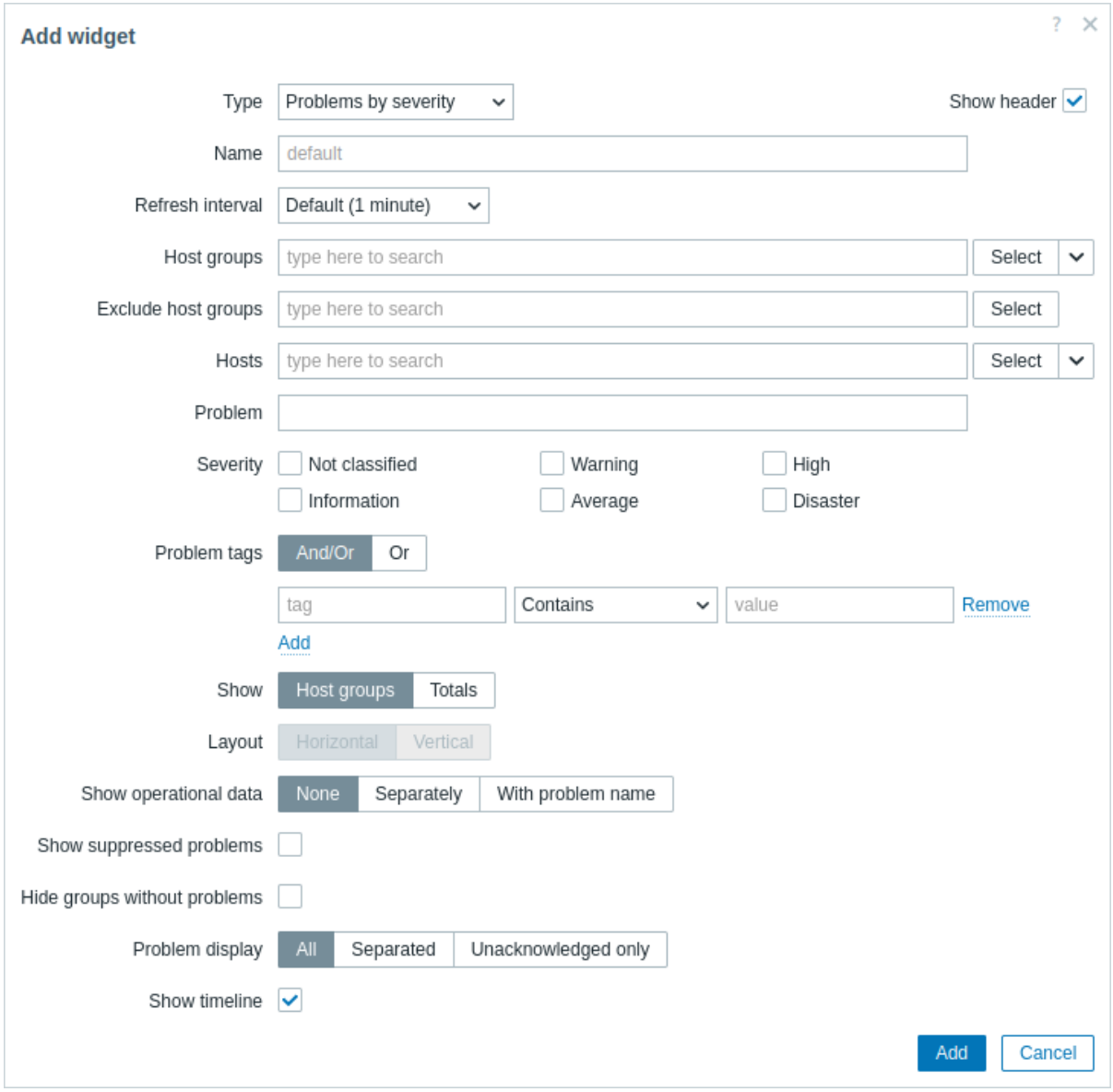


Po najechaniu kursorem na liczbę problemów wyświetlana jest lista powiązanych problemów:



Konfiguracja

Aby skonfigurować, wybierz *Problems by severity* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

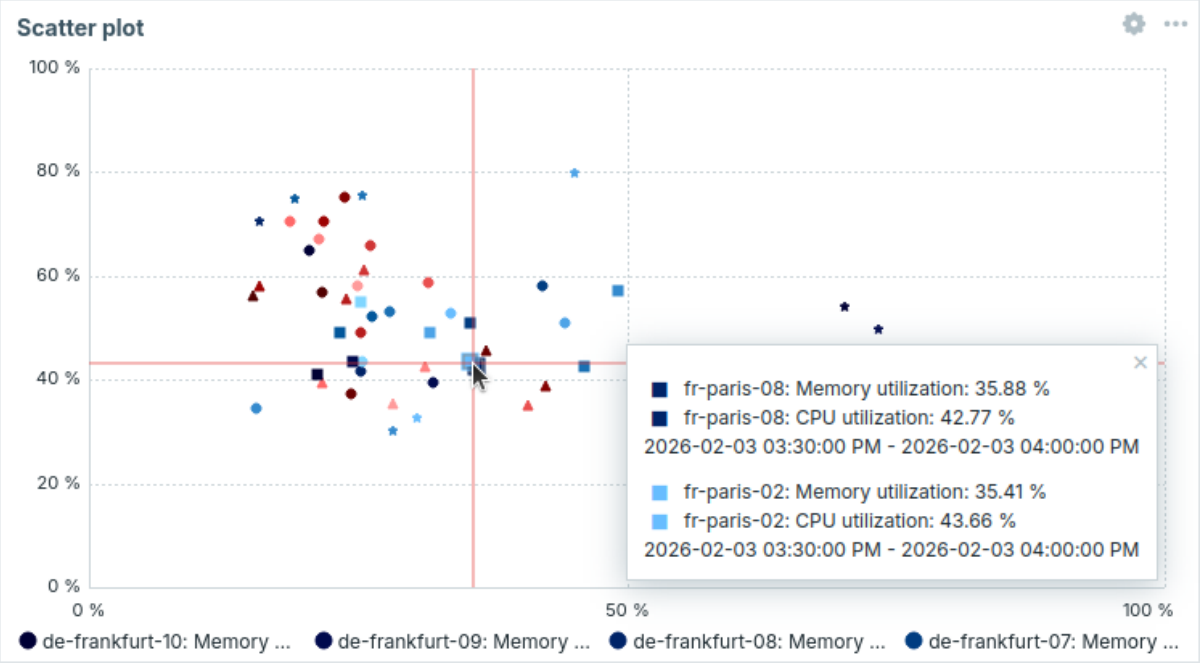
Host groups	Wybierz grupy hostów do wyświetlenia w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Dane hostów z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną podane żadne grupy hostów, wyświetlone zostaną wszystkie grupy hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.
Exclude host groups	Wybierz grupy hostów, które mają być ukryte w widżecie. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów. Dane hostów z tych grup hostów nie będą wyświetlane. Na przykład hosty 001, 002, 003 mogą znajdować się w grupie A, a hosty 002, 003 również w grupie B. Jeśli jednocześnie wybierzemy *show* grupę A i *exclude* grupę B, na dashboardzie zostaną wyświetlone tylko dane z hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.

Hosts	Wybierz hosty do wyświetlenia w widżecie. Alternatywnie wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Jeśli nie zostaną podane żadne hosty, wyświetlone zostaną wszystkie hosty. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.
Problem	Możesz ograniczyć liczbę hostów z problemami wyświetlanych według nazwy problemu. Jeśli wpiszesz tutaj ciąg znaków, wyświetlone zostaną tylko te hosty, których problemy zawierają wpisany ciąg w nazwie. Makra nie są rozwijane.
Severity	Zaznacz poziomy ważności problemów, aby odfiltrować problemy wyświetlane w widżecie. Jeśli nie zostanie zaznaczony żaden poziom ważności, wyświetlone zostaną wszystkie problemy.
Problem tags	Określ tagi problemów, aby ograniczyć liczbę problemów wyświetlanych w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wpisany ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wpisany ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek.
Show	Wybierz opcję wyświetlania: **Host groups** - wyświetl problemy według grup hostów; **Totals** - wyświetl łączną liczbę problemów dla wszystkich wybranych grup hostów w kolorowych blokach odpowiadających ważności problemu. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie, a wyświetlana będzie tylko łączna liczba problemów.
Layout	Wybierz opcję układu: **Horizontal** - kolorowe bloki sum będą wyświetlane poziomo; **Vertical** - kolorowe bloki sum będą wyświetlane pionowo. Ten parametr jest dostępny, jeśli *Show* ma ustawienie ”Totals”.
Show operational data	Wyświetl dane operacyjne jako: **None** - dane operacyjne nie będą wyświetlane **Separately** - dane operacyjne będą wyświetlane jako osobna linia **With problem name** - dane operacyjne będą wyświetlane razem z nazwą problemu. Szczegóły można znaleźć w opisie pola *Operational data* w konfiguracji wyzwalacza.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlić problemy, które w przeciwnym razie byłyby ukryte (niewyświetlane) z powodu konserwacji hosta.
Hide groups without problems	Zaznacz opcję *Hide groups without problems*, aby ukryć w widżecie dane z grup hostów bez problemów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablon dashboardzie.
Problem display	Wyświetl liczbę problemów jako: **All** - zostanie wyświetlona pełna liczba problemów; **Separated** - liczba nierozwiązanych problemów zostanie wyświetlona oddzielnie jako część łącznej liczby problemów; **Unacknowledged only** - zostanie wyświetlona tylko liczba nierozwiązanych problemów.
Show timeline	Zaznacz pole wyboru, aby wyświetlić wizualną oś czasu.

Wykres punktowy

Przegląd

Widżet *Wykres punktowy* przedstawia zależność między dwiema metrykami, nanosząc poszczególne punkty danych wzdłuż osi X i Y. Pomaga to ujawnić wzorce, skupiska, korelacje i wartości odstające w zbiorze danych.



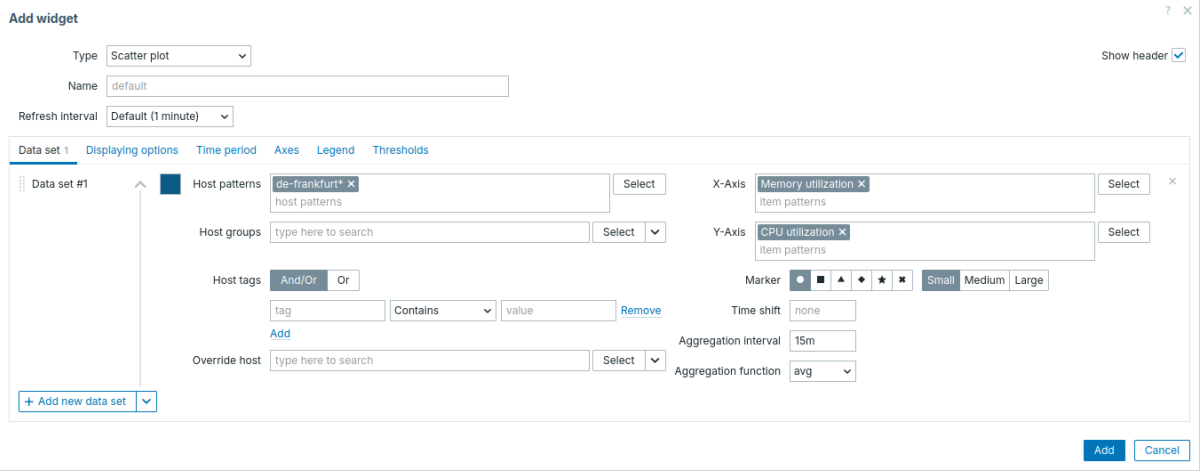
Umieszczenie wskaźnika myszy nad punktem danych wyświetla etykietę narzędzia zawierającą listę pozycji z tego punktu danych oraz innych punktów znajdujących się w promieniu 6 px; lista jest sortowana najpierw według wartości osi X wykresu punktowego w kolejności rosnącej, a następnie według wartości osi Y w kolejności malejącej.

Wybranie pozycji w etykiecie narzędzia przekazuje jej dane do innych widżetów.

Informacje wyświetlane w widżecie *Wykres punktowy* można pobrać jako obraz PNG, wybierając opcję *Pobierz obraz* w menu *Działania* widżetu ..., które pojawia się po umieszczeniu wskaźnika myszy nad widżetem. W zależności od ustawień przeglądarki obraz jest automatycznie zapisywany w folderze `Downloads` albo otwiera się okno dialogowe *Zapisz*, umożliwiające wybranie lokalizacji.

Konfiguracja

Aby skonfigurować, wybierz *Scatter plot* jako typ:



Zestaw danych

Karta **Zestaw danych** umożliwia wybór danych do wykresu punktowego poprzez dodawanie zestawów danych. Można dodać dwa typy zestawów danych:

- *Wzorce pozycji* - hosty z pasującymi pozycjami są wyświetlane jako punkty danych. Można wybrać pojedynczy kolor bazowy lub wiersz palety, aby przypisać różne kolory do każdego hosta z pasującymi pozycjami.
- *Lista pozycji* - wybrane pozycje są wyświetlane jako punkty danych, przy czym każdy punkt reprezentuje parę pozycji (X i Y). Kolor każdego zestawu danych można wybrać indywidualnie za pomocą selektora.

Punkty danych są tworzone tylko dla tych par host/pozycja, które mają co najmniej jedną wartość na obu osiach.

Domyślnie dodawany jest zestaw danych typu *Wzorce pozycji*.

Zestaw danych	Dla zestawu danych **Wzorce pozycji**: Wybierz lub wprowadź wzorce hostów i pozycji. Dane pozycji pasujących do tych wzorców zostaną wyświetlone na wykresie punktowym. Jeśli liczba punktów danych przekroczy 8 tys., należy zwiększyć wartość `memory_limit` w pliku konfiguracyjnym `php.ini`. Zalecany limit pamięci dla 20 tys. punktów danych wynosi 512 MB. Do wyboru można używać wzorców wieloznacznych (na przykład * zwróci wyniki pasujące do zera lub większej liczby znaków). Aby określić wzorzec wieloznaczny, wprowadź ciąg ręcznie i naciśnij klawisz *Enter*. Symbol wieloznaczny jest zawsze interpretowany, dlatego nie można dodać indywidualnie na przykład pozycji o nazwie *item**, jeśli istnieją inne pasujące pozycje (na przykład *item2*, *item3*). Określenie wzorców hostów i pozycji jest wymagane w przypadku zestawów danych typu „Wzorce pozycji". Zobacz także: Szczegóły konfiguracji zestawu danych. Dla zestawu danych **Lista pozycji**: Wybierz pozycje dla wykresu punktowego, klikając przycisk *Dodaj pozycję*. Można również wybrać kompatybilne widżety jako źródło danych dla pozycji, klikając przycisk *Dodaj widżet*. Określenie pozycji lub widżetów jest wymagane w przypadku zestawów danych typu „Lista pozycji". Zobacz także: Szczegóły konfiguracji zestawu danych. Dozwolone są tylko numeryczne typy pozycji. Podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu parametr określania wzorców hostów jest niedostępny, a parametr określania listy pozycji umożliwia wybór tylko pozycji skonfigurowanych w szablonie.
Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz kompatybilny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, dlatego rozpoczęcie wpisywania nazwy grupy wyświetli listę rozwijaną pasujących grup. Określenie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów zostaną wyświetlone w widżecie. Jeśli nie wprowadzono żadnych grup hostów, zostaną wyświetlone wszystkie grupy hostów. Ten parametr jest dostępny tylko dla zestawów danych typu *Wzorce pozycji* i jest niedostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Tagi hostów	Określ tagi do filtrowania hostów wyświetlanych w widżecie. Można uwzględniać lub wykluczać określone tagi i wartości tagów. Można ustawić kilka warunków. Dopasowywanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępne są następujące operatory: **Istnieje** - uwzględnia określone nazwy tagów; **Równa się** - uwzględnia określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnia określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter); **Nie istnieje** - wyklucza określone nazwy tagów; **Nie równa się** - wyklucza określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucza określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **I/Lub** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu zostaną połączone warunkiem Lub; **Lub** - wystarczy spełnienie jednego warunku. Ten parametr jest dostępny tylko dla zestawów danych typu *Wzorce pozycji* i jest niedostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.

Zastąp host	Wybierz kompatybilny widżet lub selektor hosta pulpitu nawigacyjnego jako źródło danych dla hostów. Ten parametr jest niedostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Znacznik	Wybierz typ i rozmiar znacznika dla punktów danych hostów w zestawie danych.
Przesunięcie czasu	W razie potrzeby określ przesunięcie czasu. W tym polu można używać przyrostków czasu. Dozwolone są wartości ujemne.
Funkcja agregacji	Określ funkcję agregacji, która ma być używana dla każdej pozycji w zestawie danych w ramach wybranego *Interwału agregacji*: **min** - wyświetla najmniejszą wartość; **max** - wyświetla największą wartość; **avg** - wyświetla wartość średnią; **sum** - wyświetla sumę wartości; **count** - wyświetla liczbę wartości (jeśli na obu osiach istnieje co najmniej jedna wartość); **first** - wyświetla pierwszą wartość; **last** - wyświetla ostatnią wartość. Agregacja umożliwia wyświetlenie zagregowanej wartości dla określonego interwału (5 minut, godziny, dnia) zamiast wszystkich wartości. Dla funkcji *min*, *max*, *avg* i *sum* można wyświetlać tylko dane numeryczne. W przypadku funkcji *count* dane nienumeryczne zostaną zmienione na numeryczne.
Interwał agregacji	Określ interwał agregowania wartości. W tym polu można używać przyrostków czasu. Wartość liczbowa bez przyrostka będzie traktowana jako liczba sekund. Jeśli widżet jest skonfigurowany do wyświetlania danych historycznych na podstawie trendów (*Wybór danych historycznych* jest ustawiony na *Trendy* lub *Automatycznie*), zaleca się użycie interwału agregacji będącego wielokrotnością 1 godziny (np. 3600, 60m, 1h, 3h itd.). Trendy przechowują zagregowane wartości godzinowe, dlatego użycie interwału agregacji, który nie jest wielokrotnością 1 godziny (np. 100s, 7min, 15min, 90min itd.), może prowadzić do wyników trudnych do zinterpretowania. Określenie interwału agregacji jest wymagane. Interwał agregacji określa przedział czasu i współrzędne punktu danych. Na przykład jeśli pozycje osi X mają 4 wartości, a pozycje osi Y mają 1 wartość, współrzędne odzwierciedlają agregację 4 wartości na osi X i 1 wartości na osi Y w określonym interwale.

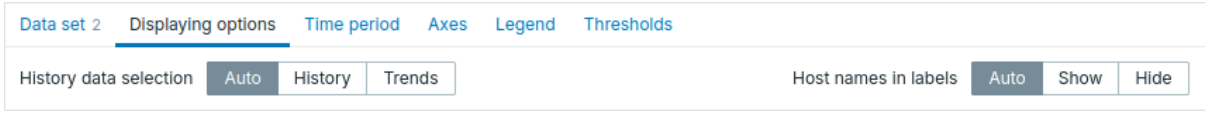
Szczegóły konfiguracji zestawu danych

Istniejące zestawy danych są wyświetlane na liście. Możesz zmieniać ich kolejność, rozwijać/zwijać, zmieniać kolory oraz klonować te zestawy danych.

Więcej informacji można znaleźć w opisie szczegółów konfiguracji zestawu danych w widżecie *Graph*. Te szczegóły mają również zastosowanie do widżetu *Scatter plot*.

Opcje wyświetlania

Karta **Opcje wyświetlania** umożliwia zdefiniowanie wyboru danych historycznych:

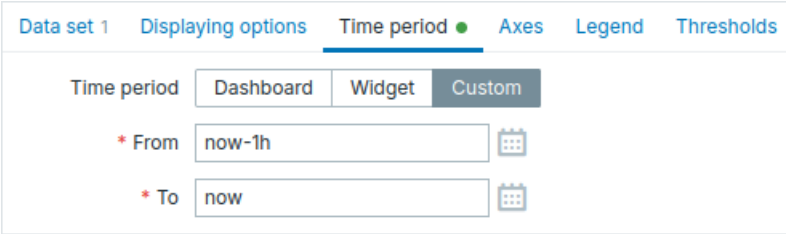


Wybór danych historycznych	Ustaw źródło danych wykresu punktowego: **Automatycznie** - dane są pobierane zgodnie z algorytmem klasycznego wykresu (domyślnie). **Historia** - dane z historii. **Trendy** - dane z trendów.

Nazwy hostów na etykietach	Wybierz, czy wyświetlać nazwy hostów w legendzie wykresu oraz w etykiecie narzędzia wyświetlanej po najechaniu kursorem na dane wykresu: **Automatycznie** - nazwy hostów są wyświetlane tylko wtedy, gdy w zestawach danych występuje więcej niż jeden host (domyślnie). **Pokaż** - nazwy hostów są wyświetlane. **Ukryj** - nazwy hostów są ukryte. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.

Okres

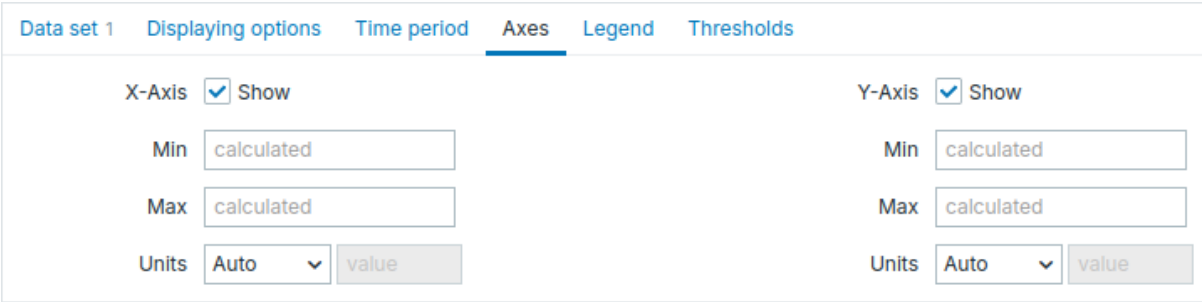
Karta **Okres** umożliwia ustawienie okresu, dla którego mają być wyświetlane dane na wykresie punktowym:



Okres	Wybierz źródło danych dla okresu: **Dashboard** - użyj selektora okresu dashboardu; **Widget** - użyj kompatybilnego widgetu (ustawionego w parametrze *Widget*); **Niestandardowy** - użyj niestandardowego okresu ustawionego w parametrach *Od* i *Do*; jeśli jest ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara wskazująca ustawiony czas po najechaniu kursorem myszy. Należy pamiętać, że niezależnie od konfiguracji *Okresu* widgetu kompatybilne widgety nadal mogą używać go jako źródła danych dla okresu.
Widget	Wprowadź lub wybierz kompatybilny widget jako źródło danych dla okresu. Ten parametr jest dostępny, jeśli dla opcji *Okres* ustawiono wartość „Widget”.
Od	Wprowadź lub wybierz początek okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla opcji *Okres* ustawiono wartość „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla opcji *Okres* ustawiono wartość „Niestandardowy”.

Osie

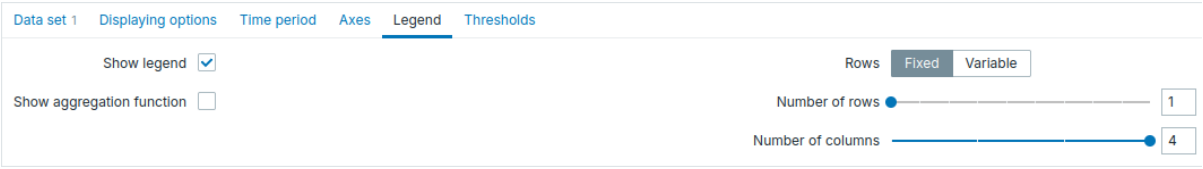
Karta **Osie** umożliwia dostosowanie sposobu wyświetlania osi:



Oś X/Oś Y	Odznacz to pole wyboru, aby ukryć oś X/oś Y (domyślnie zaznaczone).
Min	Ustaw minimalną wartość odpowiedniej osi.
Max	Ustaw maksymalną wartość odpowiedniej osi.
Jednostki	Wybierz jednostkę dla wartości osi wykresu punktowego z listy rozwijanej: **Auto** - wartości osi są wyświetlane przy użyciu jednostki pierwszej pozycji w zestawie danych; **Statyczne** - wartości osi są wyświetlane przy użyciu jednostki określonej w polu wejściowym *value*; jeśli pole pozostanie puste, wyświetlane są tylko wartości liczbowe.

Legenda

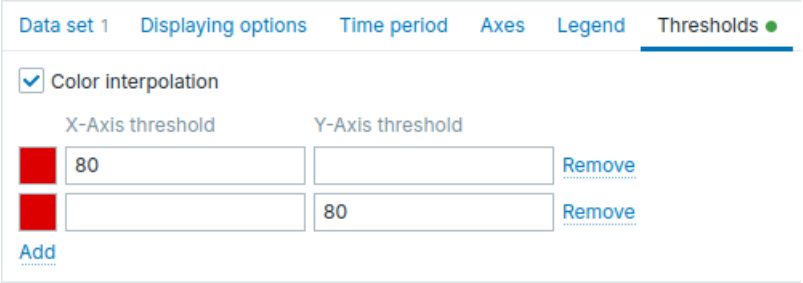
Karta **Legenda** umożliwia dostosowanie legendy wykresu punktowego:



Pokaż legendę	Odznacz to pole wyboru, aby ukryć legendę na wykresie punktowym (domyślnie zaznaczone).
Pokaż funkcję agregacji	Zaznacz to pole wyboru, aby wyświetlić funkcję agregacji w legendzie.
Wiersze	Wybierz tryb wyświetlania wierszy legendy: **Stała** - liczba wyświetlanych wierszy jest określana przez wartość parametru *Liczba wierszy*; **Zmienna** - liczba wyświetlanych wierszy jest określana przez liczbę skonfigurowanych pozycji, ale nie może przekroczyć wartości parametru *Maksymalna liczba wierszy*.
Liczba wierszy/ Maksymalna liczba wierszy	Jeśli dla opcji *Wiersze* ustawiono wartość „Stała”, ustaw liczbę wyświetlanych wierszy legendy (1-10). Jeśli dla opcji *Wiersze* ustawiono wartość „Zmienna”, ustaw maksymalną liczbę wyświetlanych wierszy legendy (1-10).
Liczba kolumn	Ustaw liczbę wyświetlanych kolumn legendy (1-4). Ten parametr jest dostępny, jeśli opcja *Wyświetl min./śr./maks.* jest odznaczona.

Progi

Karta **Progi** umożliwia skonfigurowanie progów określających kolory znaczników hostów:



Interpolacja kolorów	Zaznacz pole wyboru, aby włączyć płynne przechodzenie między kolorami progów dla znaczników hostów. Ten parametr ma zastosowanie, jeśli ustawiono co najmniej dwa progi dla osi X, osi Y lub obu osi jednocześnie.
Próg osi X/osi Y	Kliknij *Dodaj*, aby dodać próg, wybierz kolor progu za pomocą selektora kolorów i określ wartość liczbową. Progi można stosować do jednej osi, do obu osi osobno lub do obu osi jednocześnie. Na przykład pojedynczy próg z wartościami X=80 i Y=80 pokoloruje znaczniki, gdy oba warunki zostaną spełnione. Lista progów zostanie posortowana rosnąco podczas zapisywania. Obsługiwane są przyrostki (na przykład „1d”, „2w”, „4K”, „8G”). Obsługiwane są mapowania wartości.

Raport SLA

Przegląd

Widżet *Raport SLA* wyświetla raporty SLA, pomagając monitorować wydajność usług względem zdefiniowanych celów.

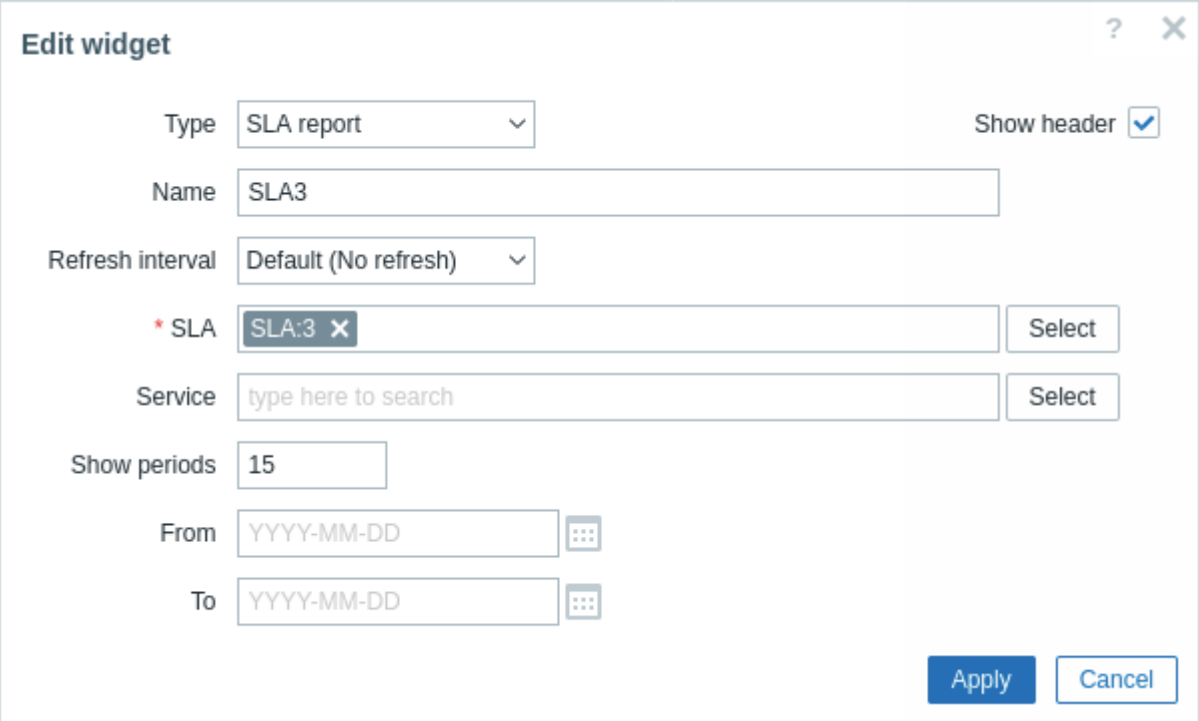
SLA report

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

Wyświetla te same dane co *Usługi* > *Raport SLA*.

Konfiguracja

Aby skonfigurować widget, wybierz *SLA report* jako typ:



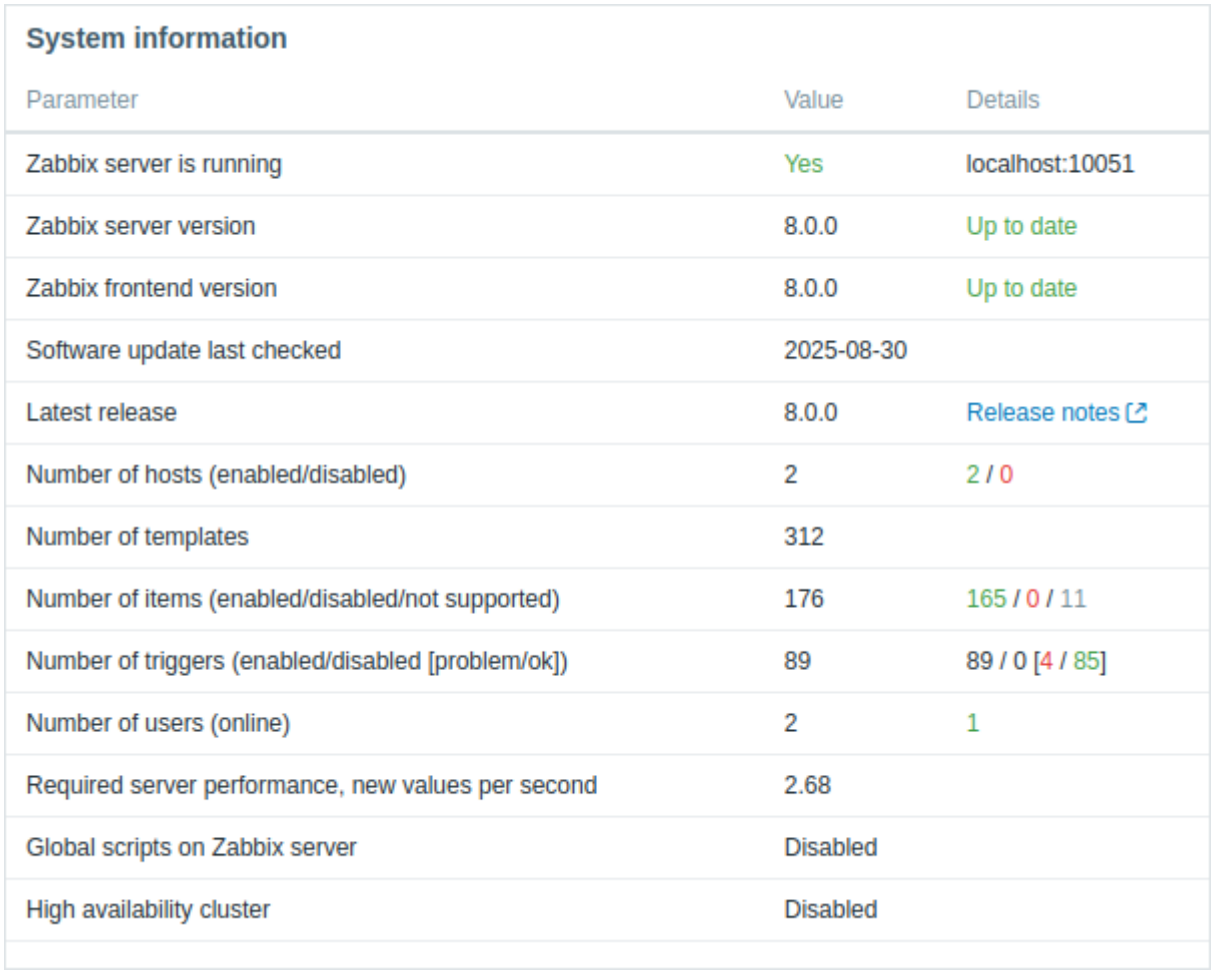
Oprócz parametrów wspólnych dla wszystkich widgetów można ustawić następujące opcje specyficzne:

SLA	Wybierz SLA dla raportu.
Service	Wybierz usługę dla raportu.
Show periods	Ustaw, ile okresów będzie wyświetlanych w widżecie (domyślnie 20, maksymalnie 100).
From	Wybierz datę początkową raportu. Obsługiwane są daty względne: `now`, `now/d`, `now/w-1w` itd.; obsługiwane modyfikatory daty: d, w, M, y.
To	Wybierz datę końcową raportu. Obsługiwane są daty względne: `now`, `now/d`, `now/w-1w` itd.; obsługiwane modyfikatory daty: d, w, M, y.

Informacje o systemie

Przegląd

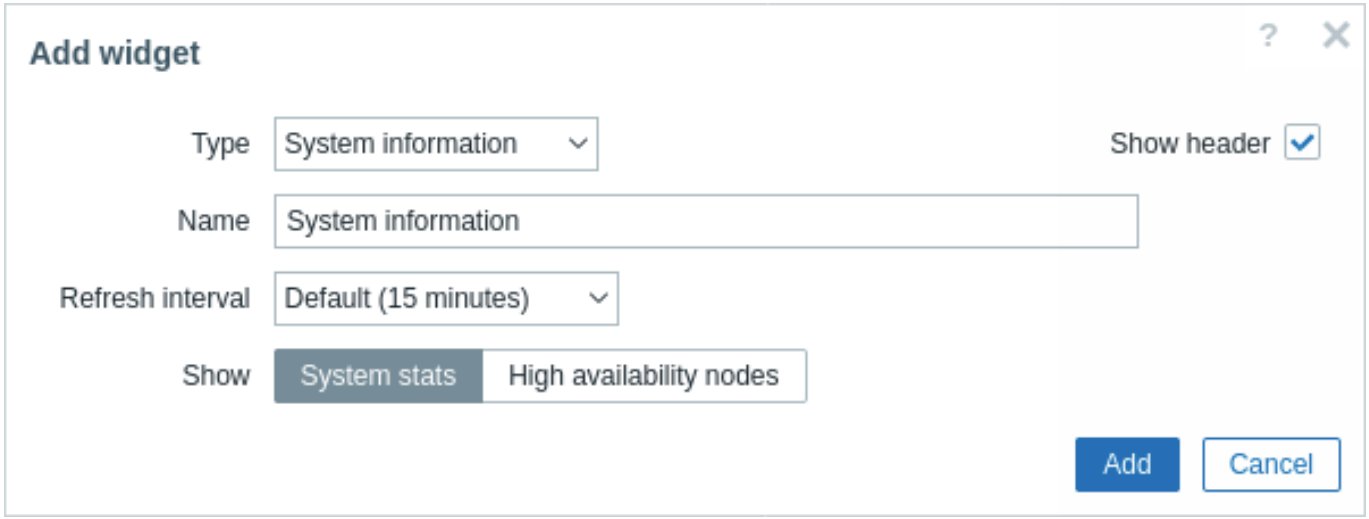
Widżet *Informacje o systemie* wyświetla podsumowanie kluczowych danych serwera Zabbix i systemu, a gdy włączona jest wysoka dostępność, także szczegóły węzłów wysokiej dostępności.



Wyświetla te same dane co *Raporty* > *Informacje o systemie*, z wyjątkiem *Zabbix server ID*.

Konfiguracja

Aby skonfigurować, wybierz *Informacje o systemie* jako typ:



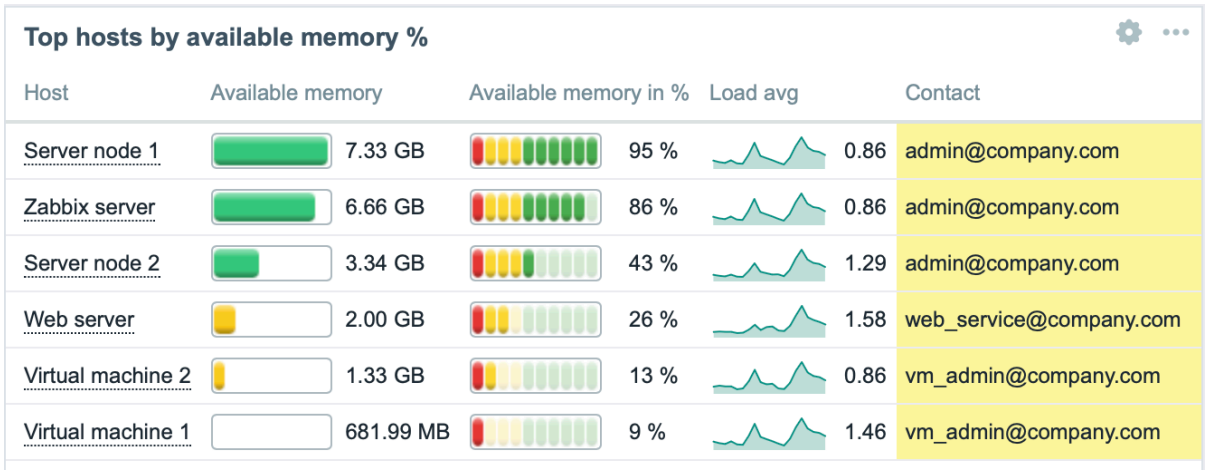
Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje:

Pokaż	Wybierz, co ma być wyświetlane: **Statystyki systemu** - wyświetla podsumowanie najważniejszych danych serwera Zabbix i systemu; **Węzły wysokiej dostępności** - wyświetla stan węzłów wysokiej dostępności (jeśli włączono klaster wysokiej dostępności).
Pokaż szczegóły sprawdzania aktualizacji oprogramowania	Zaznacz pole wyboru, aby wyświetlać szczegóły sprawdzania aktualizacji oprogramowania Zabbix. Ta opcja jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix i w polu *Pokaż* wybrano opcję „Statystyki systemu”.

Najlepsze hosty

Przegląd

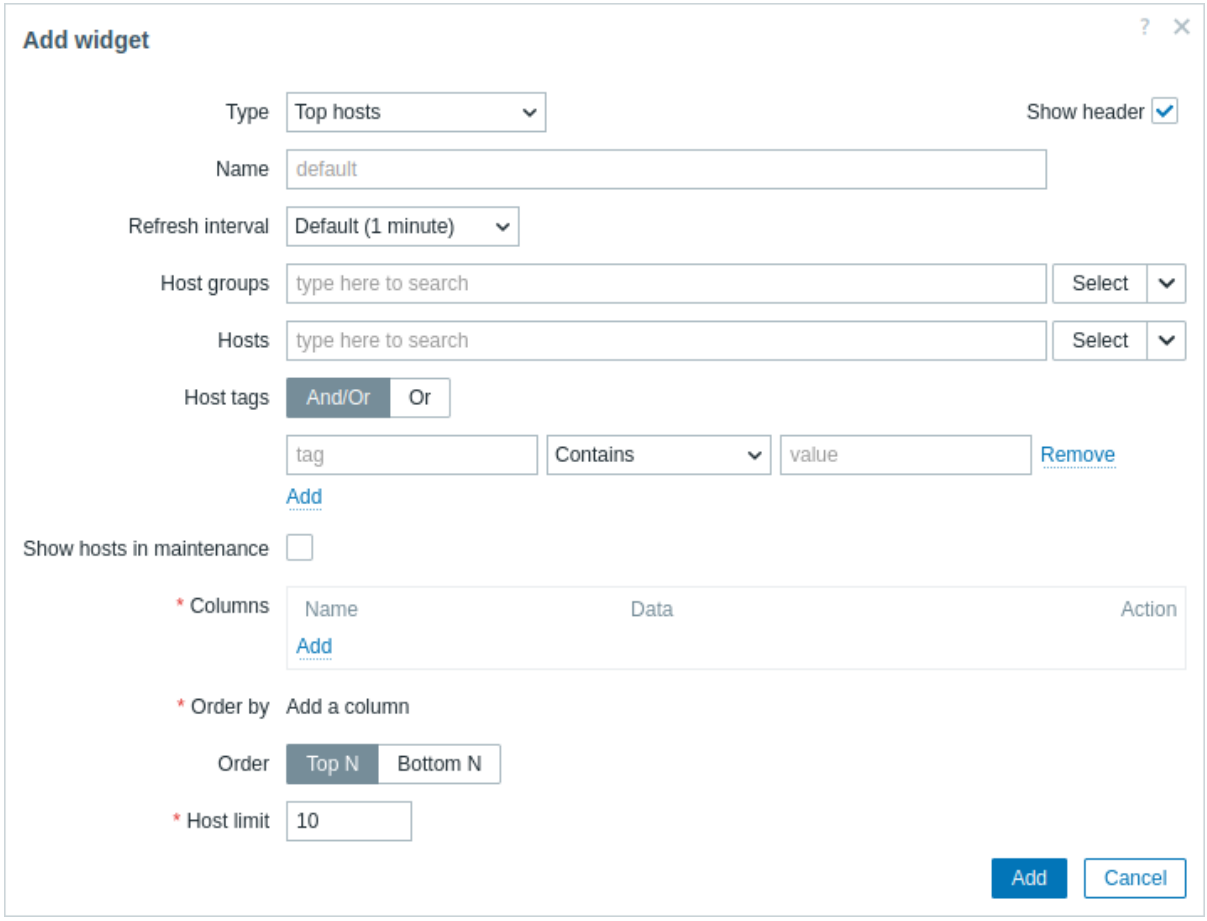
Widżet *Top hosts* wyświetla najwyższe/najniższe wartości dla wybranych pozycji z wielu hostów.



Widżet obsługuje jednoczesne wyświetlanie do 1000 hostów.

Konfiguracja

Aby skonfigurować, wybierz *Top hosts* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów można ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Hosts	Wybierz hosty, które mają być wyświetlane w widżecie. Alternatywnie wybierz zgodny widget lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Host tags	Określ tagi, aby ograniczyć liczbę hostów wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** - wystarczy spełnienie jednego warunku. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Show hosts in maintenance	Zaznacz to pole wyboru, aby wyświetlać również hosty w utrzymaniu (w takim przypadku obok nazwy hosta zostanie pokazana ikona utrzymania). Domyślnie niezaznaczone.

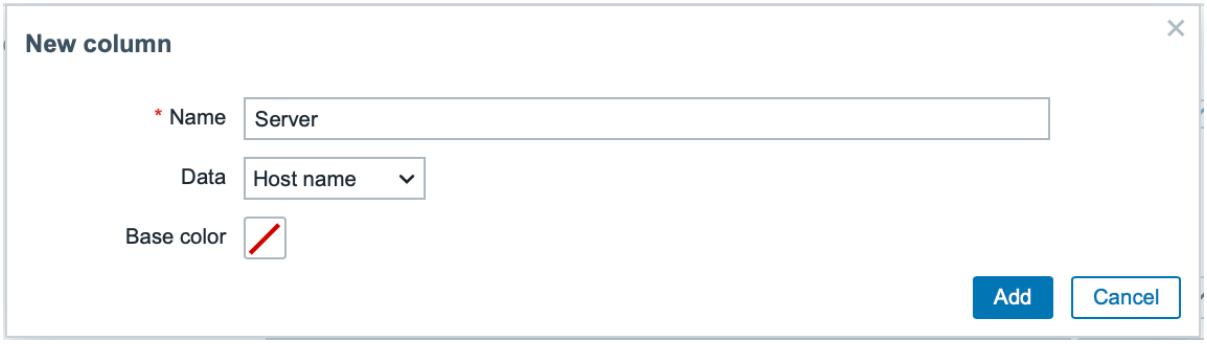
Columns	Dodaj kolumny danych do wyświetlenia. Kolejność kolumn określa ich wyświetlanie od lewej do prawej. Kolumny można zmieniać miejscami, przeciągając je w górę i w dół za uchwyt znajdujący się przed nazwą kolumny.
Order by	Określ kolumnę ze zdefiniowanej listy *Columns*, która ma być używana do sortowania *Top N* lub *Bottom N*.
Order	Określ kolejność wierszy: **Top N** - malejąco zgodnie z zagregowaną wartością *Order by*; **Bottom N** - rosnąco zgodnie z zagregowaną wartością *Order by*.
Host limit	Liczba wierszy hostów do wyświetlenia (1-1000). Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

Konfiguracja kolumn

Możliwe jest dodanie kolumn trzech typów danych: nazwa hosta, wartość pozycji lub tekst. Lista dostępnych parametrów kolumny zależy od typu danych kolumny, a w przypadku typu wartość pozycji — od formatu wartości.

Kolumna Nazwa hosta

Kolumna Nazwa hosta służy do wyświetlania nazwy hosta.

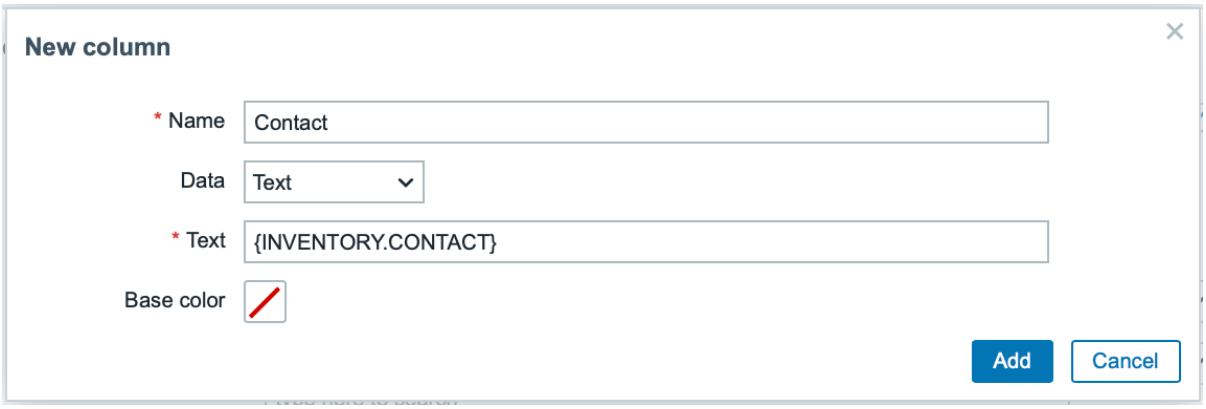


Obsługiwane parametry:

Name	Nazwa kolumny.
Data	Typ danych wyświetlanych w kolumnie; wybierz **Nazwa hosta**.
Base color	Kolor tła kolumny.

Kolumna tekstowa

Kolumna tekstowa służy do wyświetlania dowolnego określonego ciągu tekstowego.

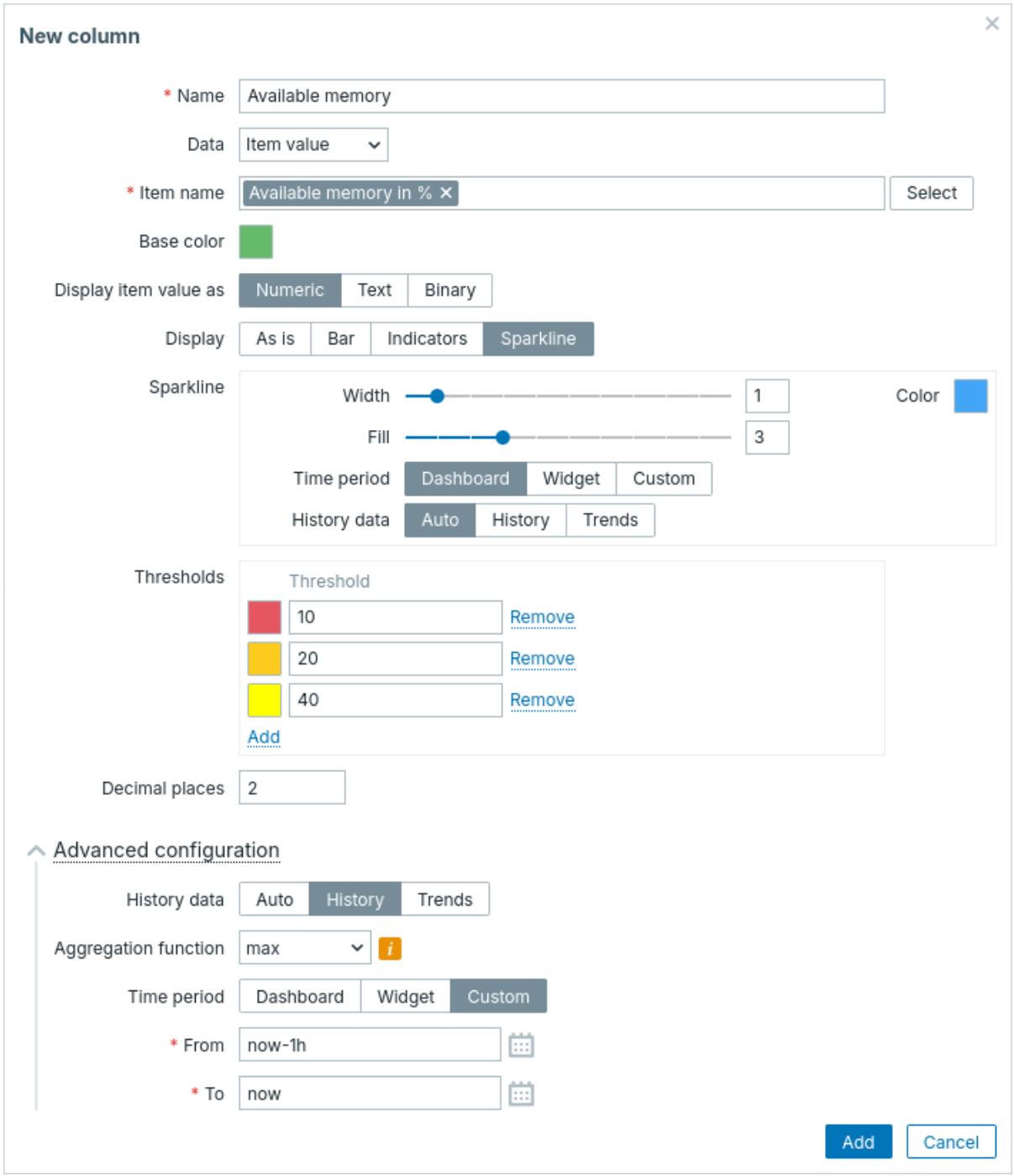


Obsługiwane parametry:

Nazwa	Nazwa kolumny.
Dane	Typ danych wyświetlany w kolumnie, wybierz **Tekst**.
Kolor bazowy	Kolor tła kolumny.
Tekst	Wprowadź ciąg do wyświetlenia. Może zawierać makra hosta i inwentarza.

Kolumna wartości pozycji

Kolumna wartości pozycji służy do wyświetlania wartości określonej pozycji; obsługuje wiele formatów i opcji wyświetlania wartości.



Obsługiwane parametry:

Name	Nazwa kolumny.
Data	Typ danych wyświetlanych w kolumnie; wybierz **Item value**.
Item name	Wybierz pozycję; jej nazwa jest używana do dopasowania i wyświetlania pozycji o tej samej nazwie na wszystkich wybranych hostach. Jeśli host ma wiele takich pozycji, widget wyświetla pozycję, której klucz znajduje się jako pierwszy w kolejności alfabetycznej (np. *CPU utilization* z kluczem `proc.cpu.util` zamiast `system.cpu.util`). Podczas konfigurowania widgetu na dashboardzie szablonu można wybrać tylko pozycje skonfigurowane w szablonie.
Base color	Kolor tła kolumny; kolor wypełnienia, jeśli dane *Item value* są wyświetlane jako pasek lub wskaźniki. Domyślny kolor może zostać zastąpiony kolorem niestandardowym, jeśli wartość pozycji przekracza jeden z określonych progów.
Display item value as	Format wyświetlania wartości pozycji: **Numeric**, **Text** lub **Binary**. Wybrana opcja określa, które dodatkowe parametry będą dostępne. Zapoznaj się z listą parametrów dla każdego formatu poniżej.

Advanced configuration	
History data	Pobieranie danych z historii lub trendów: **Auto** - wybór automatyczny; **History** - pobieranie danych historycznych; **Trends** - pobieranie danych trendów. Ten parametr jest dostępny tylko dla numerycznych wartości pozycji. W przypadku wartości tekstowych i binarnych dane będą zawsze pobierane z historii.
Aggregation function	Określ funkcję agregacji używaną w wybranym *Time period*: **min** - wyświetlanie najmniejszej wartości; **max** - wyświetlanie największej wartości; **avg** - wyświetlanie wartości średniej; **count** - wyświetlanie liczby wartości; **sum** - wyświetlanie sumy wartości; **first** - wyświetlanie pierwszej wartości; **last** - wyświetlanie ostatniej wartości. Jeśli wybrano **not used**, wyświetlana jest najnowsza wartość (bez agregacji). Agregacja umożliwia wyświetlenie zagregowanej wartości dla określonego przedziału (5 minut, godziny, dnia) zamiast najnowszej wartości. Dla funkcji *min*, *max*, *avg* i *sum* można wyświetlać tylko dane numeryczne. W przypadku funkcji *count* dane nienumeryczne zostaną zmienione na numeryczne.
Time period	Określ przedział czasu używany do agregowania wartości. Wybierz źródło danych dla przedziału czasu: **Dashboard** - użyj selektora przedziału czasu dashboardu; **Widget** - użyj zgodnego widgetu (ustawionego w parametrze *Widget*); **Custom** - użyj niestandardowego przedziału czasu ustawionego w parametrach *From* i *To*; jeśli zostanie ustawiony, w prawym górnym rogu widgetu będzie wyświetlana ikona zegara wskazująca ustawiony czas po najechaniu kursorem. Należy pamiętać, że niezależnie od konfiguracji *Time period* widgetu zgodne widgety nadal mogą używać go jako źródła danych dla przedziału czasu. Ten parametr nie jest dostępny, jeśli dla *Aggregation function* ustawiono wartość „not used”.
Widget	Wprowadź lub wybierz zgodny widget jako źródło danych dla przedziału czasu. Ten parametr jest dostępny, jeśli dla *Time period* ustawiono wartość „Widget”.
From	Wprowadź lub wybierz początek przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Time period* ustawiono wartość „Custom”.
To	Wprowadź lub wybierz koniec przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Time period* ustawiono wartość „Custom”.

Parametry **Numeric item value**:

Display	Określ sposób wyświetlania wartości: **As is** - jako zwykły tekst; **Bar** - jako jednolity pasek wypełniony kolorem; **Indicators** - jako segmentowany pasek wypełniony kolorem; **Sparkline** - jako miniaturowy wykres liniowy.
Thresholds	Określ wartości progowe, przy których powinien zmienić się kolor tła lub wypełnienia. Po zapisaniu lista zostanie posortowana rosnąco. W przypadku wykresów sparkline progi są stosowane tylko do ostatniej wartości pozycji.
Decimal places	Określ liczbę miejsc dziesiętnych wyświetlanych wraz z wartością.
Bar/Indicators	
Min	Wartość minimalna.
Max	Wartość maksymalna.
Sparkline	
Width	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10.
Color	Wybierz kolor linii i wypełnienia.
Fill	Ustaw poziom przezroczystości koloru wypełnienia za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10.

Time period	Określ przedział czasu, z którego wartości mają zostać uwzględnione na wykresie sparkline. Wybierz źródło danych dla przedziału czasu: **Dashboard** - użyj selektora przedziału czasu dashboardu **Widget** - użyj zgodnego widgetu (ustawionego w parametrze *Widget*); **Custom** - użyj niestandardowego przedziału czasu ustawionego w parametrach *From* i *To*; jeśli zostanie ustawiony, w prawym górnym rogu widgetu będzie wyświetlana ikona zegara wskazująca ustawiony czas po najechaniu kursorem. Należy pamiętać, że niezależnie od konfiguracji *Time period* widgetu zgodne widgety nadal mogą używać go jako źródła danych dla przedziału czasu.
Widget	Wprowadź lub wybierz zgodny widget jako źródło danych dla przedziału czasu. Ten parametr jest dostępny, jeśli dla *Time period* ustawiono wartość „Widget”.
From	Wprowadź lub wybierz początek przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Time period* ustawiono wartość „Custom”.
To	Wprowadź lub wybierz koniec przedziału czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Time period* ustawiono wartość „Custom”.
History data	Pobieranie danych z historii lub trendów: **Auto** - wybór automatyczny; **History** - pobieranie danych historycznych; **Trends** - pobieranie danych trendów.

Parametry **Text item value**:

Highlights	Określ wyrażenia regularne, po dopasowaniu których powinien zmienić się kolor tła lub wypełnienia.

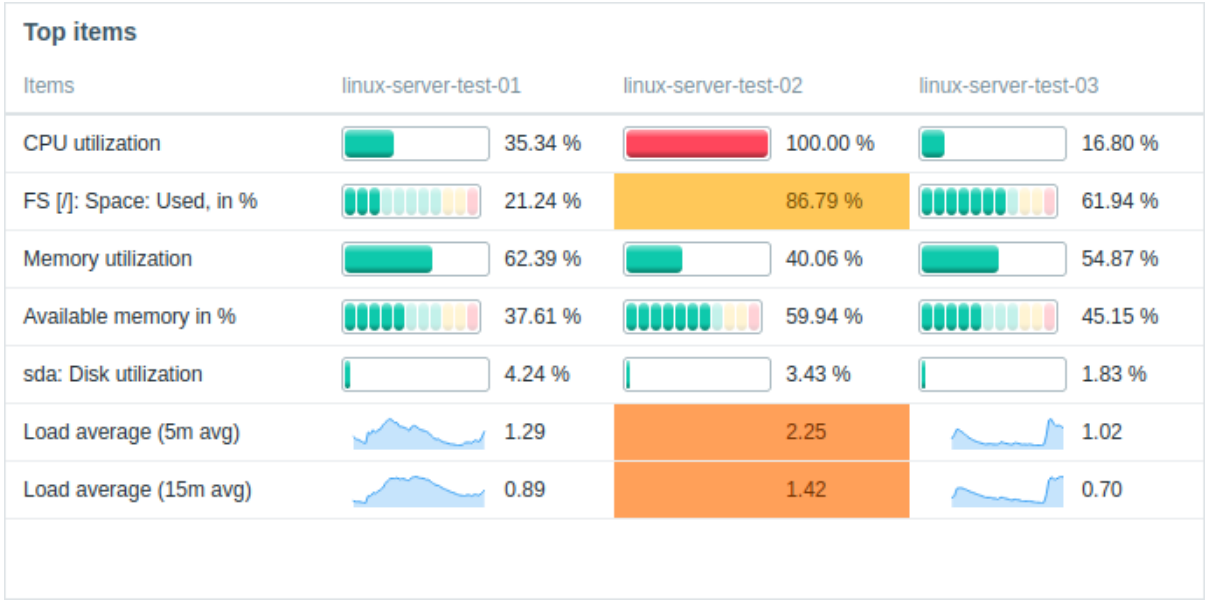
Parametry **Binary item value**:

Show thumbnail	Określ, czy utworzyć i wyświetlić miniaturę obrazu zawierającego dane binarne, czy wyświetlić hiperłącze *Show* prowadzące do obrazu w pełnym rozmiarze w kolumnie wartości.

Najważniejsze pozycje

Przegląd

Widżet *Top items* wyświetla najwyższe/najniższe wartości dla wybranych pozycji, zapewniając szybki przegląd ich wydajności.

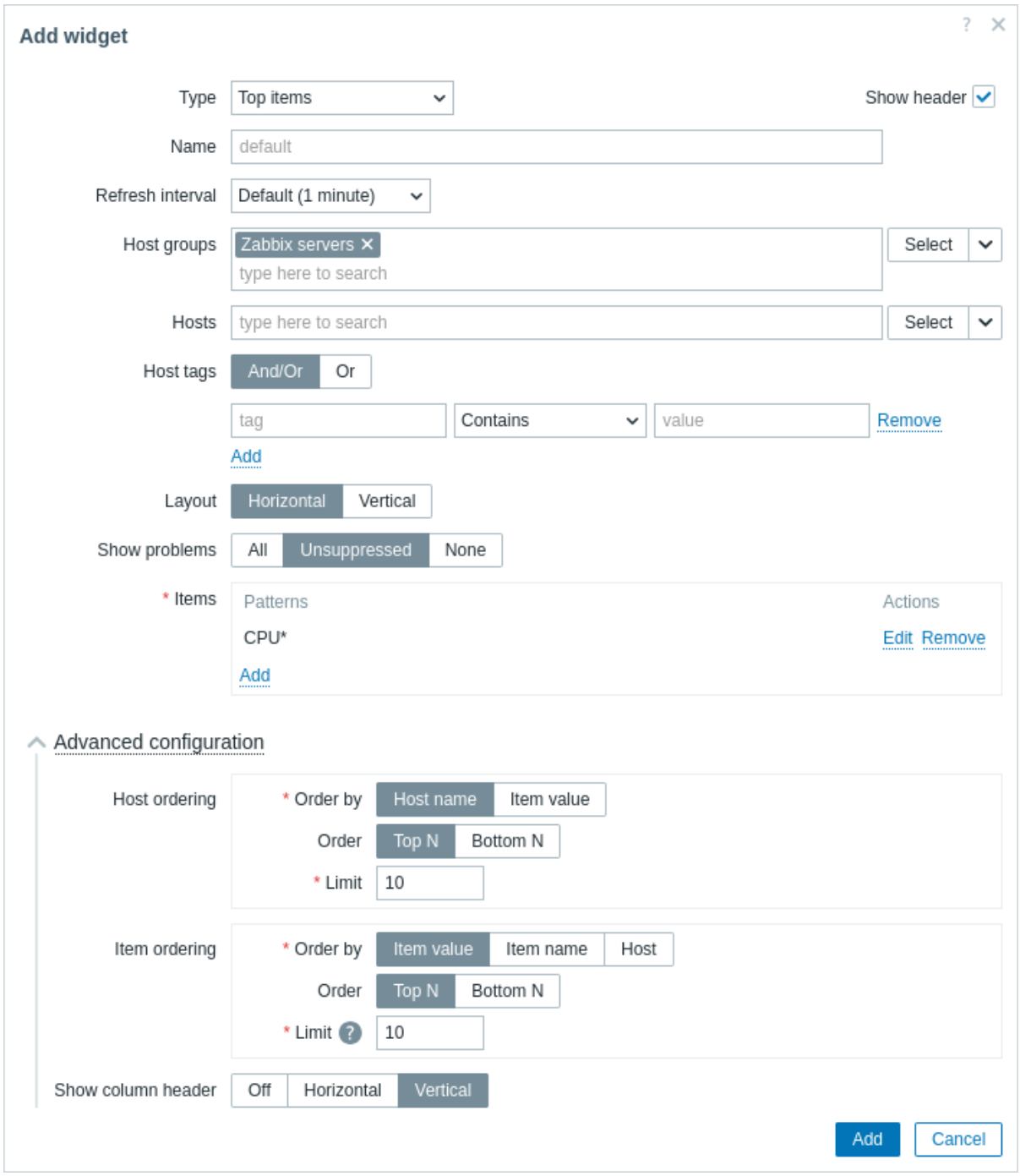


Domyślnie widżet wyświetla wartości z ostatnich 24 godzin. Ten okres można zmienić w *Administracja* > *Ogólne* > *GUI* (*Maksymalny okres wyświetlania historii*).

Kliknięcie wartości pozycji otwiera menu pozycji.

Konfiguracja

Aby skonfigurować, wybierz *Top items* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

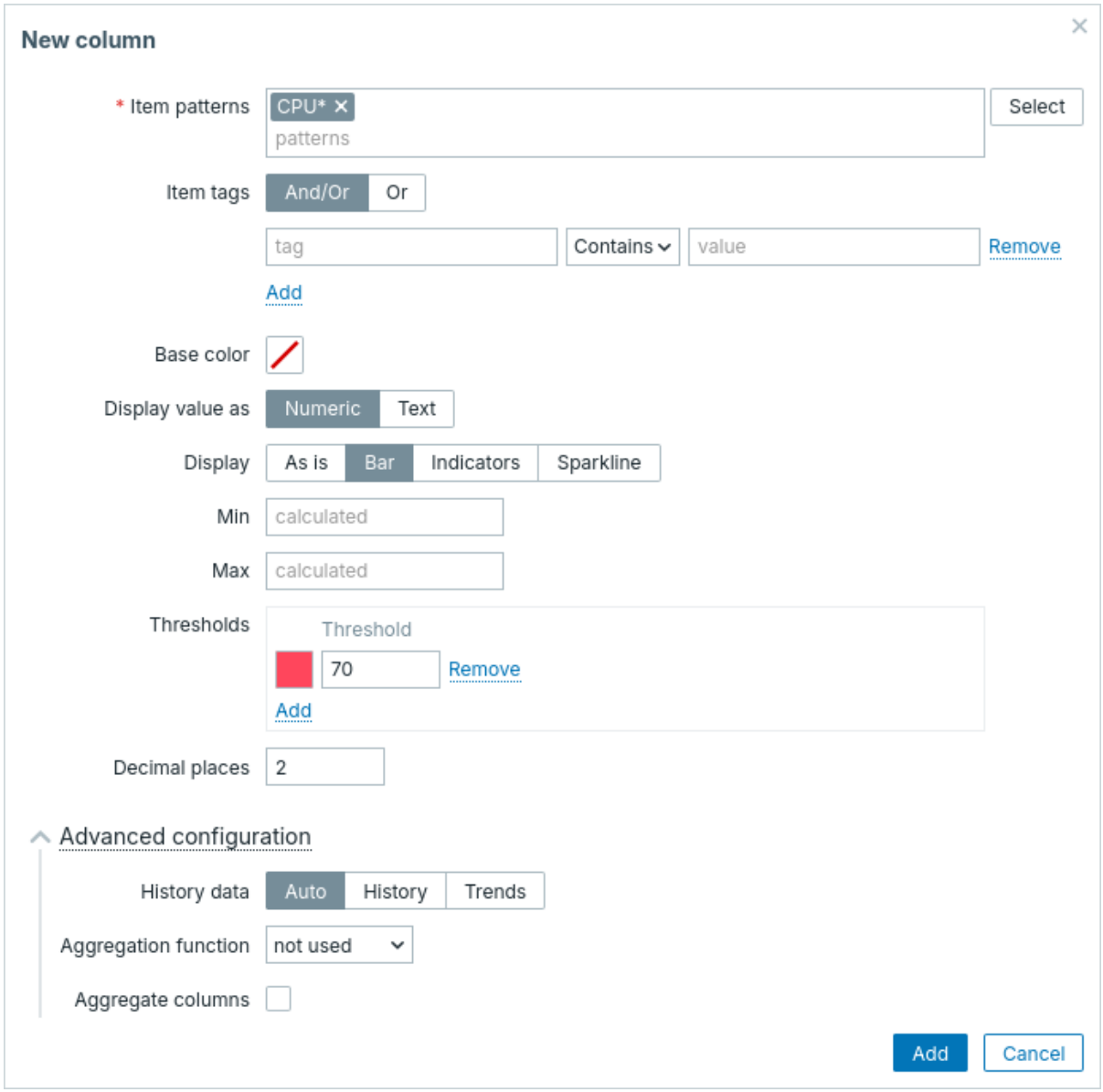
Host groups	Wybierz grupy hostów. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy grupy spowoduje wyświetlenie listy rozwijanej pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.
Hosts	Wybierz hosty. Alternatywnie wybierz zgodny widżet lub dashboard jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc rozpoczęcie wpisywania nazwy hosta spowoduje wyświetlenie listy rozwijanej pasujących hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym dashboardzie.

Host tags		Określ tagi, aby ograniczyć liczbę danych hostów wyświetlanych w widżecie. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (uwzględnia wielkość liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie fragmentu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (uwzględnia wielkość liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie fragmentu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku *Or*; **Or** - wystarczy, że spełniony będzie jeden warunek.
Layout		Wybierz opcję układu: **Horizontal** - nazwy hostów będą wyświetlane po lewej stronie; **Vertical** - nazwy hostów będą wyświetlane u góry.
Show problems		Filtruj problemy do wyświetlenia na podstawie ich statusu (all, unsuppressed, none). Problemy są oznaczane kolorowymi wartościami pozycji, zastępując ustawienia wyświetlania kolumny pozycji. Kolory są oparte na ważności problemu, którą można dostosować w *Administration* > *General* > *Trigger displaying options*. Ważność problemów można zmieniać podczas aktualizowania problemów.
Items		Dodaj wzorce pozycji lub konkretne pozycje do wyświetlenia (zobacz Konfiguracja kolumn).
Advanced configuration		
Host ordering		Wybierz opcje sortowania dla kolumny/wiersza hostów.
	Order by	Sortuj hosty według: **Host name** - hosty będą sortowane według nazwy hosta; **Item value** - hosty będą sortowane według wartości wybranych pozycji.
	Order	Wybierz, czy wyświetlać najwyższe, czy najniższe wartości: **TopN** - N najwyższych wartości; **BottomN** - N najniższych wartości. Wartość N jest wybierana w polu *Limit*.
	Limit	Wprowadź limit dla hostów możliwych do wyświetlenia (zakres 1-1000; domyślnie 10). Ta wartość stanie się wartością N w polu *Order*.
Item ordering		Wybierz opcje sortowania dla kolumny/wiersza pozycji.
	Order by	Sortuj pozycje według: **Item value** - pozycje będą sortowane według wartości pozycji; **Item name** - pozycje będą sortowane według nazwy pozycji; **Host** - pozycje będą sortowane według wybranego wzorca hosta.
	Order	Wybierz, czy wyświetlać najwyższe, czy najniższe wartości: **TopN** - N najwyższych wartości; **BottomN** - N najniższych wartości. Wartość N jest wybierana w polu *Limit*.
	Limit	Wprowadź limit dla pozycji możliwych do wyświetlenia (zakres 1-1000; domyślnie 10). Ta wartość stanie się wartością N w polu *Order*.

Show column header	Wybierz opcje wyświetlania nagłówka kolumny: **Off** - nie wyświetlaj nagłówka kolumny; **Horizontal** - wyświetlaj tekst poziomo w nagłówku; **Vertical** - wyświetlaj tekst pionowo w nagłówku.

Konfiguracja kolumn

Aby skonfigurować kolumny pozycji, kliknij *Dodaj* w parametrze *Pozycje*:



Wspólne parametry kolumn:

Wzorce pozycji	Określ jeden lub kilka wzorców pozycji, używając znaku wieloznacznego. Alternatywnie wybierz pozycje. Podczas konfigurowania widżetu na pulpicie szablonu można wybrać tylko pozycje skonfigurowane w szablonie.

Tagi pozycji	Określ tagi, aby ograniczyć liczbę danych pozycji wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze rozróżnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, w których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, w których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane przez warunek *Or*; **Or** - wystarczy spełnienie jednego warunku.
Kolor bazowy	Wybierz kolor tła kolumny lub kolor wypełnienia, jeśli dla parametru *Wyświetlanie* ustawiono wartość „Słupek” lub „Wskaźniki”. Zwróć uwagę, że kolor bazowy może zostać zastąpiony przez kolory progów lub wyróżnienia.
Wyświetlaj wartość jako	Format wyświetlania wartości pozycji: **Liczbowy** lub **Tekst**. Wybrana opcja określa, które dodatkowe parametry będą dostępne. Szczegóły znajdują się na poniższej liście parametrów dla każdego formatu.

Parametry kolumn dla wartości liczbowych:

Wyświetlanie	Określ sposób wyświetlania wartości: **Tak jak jest** - jako zwykły tekst; **Pasek** - jako jednolity pasek wypełniony kolorem; **Wskaźniki** - jako segmentowany pasek wypełniony kolorem; **Sparkline** - miniwykres liniowy.
Min	Minimalna wartość dla wyświetlania pasków/wskaźników.
Maks.	Maksymalna wartość dla wyświetlania pasków/wskaźników.
Szerokość	Ustaw grubość linii wykresu za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10. Ten parametr dotyczy tylko wyświetlania sparkline.
Wypełnienie	Ustaw poziom przezroczystości wypełnienia za pomocą suwaka lub ręcznie, wprowadzając wartość z zakresu od 0 do 10. Ten parametr dotyczy tylko wyświetlania sparkline.
Kolor	Wybierz kolor linii i wypełnienia. Ten parametr dotyczy tylko wyświetlania sparkline.
Opis pól związanych z wyborem okresu i danych historycznych znajduje się w sekcji *Konfiguracja zaawansowana*.	
Progi	Określ wartości progowe, przy których powinien zmienić się kolor tła/wypełnienia. Po zapisaniu lista zostanie posortowana rosnąco. W przypadku sparkline progi są stosowane tylko do ostatniej wartości pozycji.
Miejsca dziesiętne	Określ, ile miejsc dziesiętnych ma być wyświetlanych wraz z wartością.
Konfiguracja zaawansowana	
Dane historyczne	Pobieraj dane z historii lub trendów: **Automatycznie** - wybór automatyczny; **Historia** - pobieraj dane historyczne; **Trendy** - pobieraj dane trendów. Ten parametr jest dostępny tylko dla wartości liczbowych pozycji. W przypadku wartości tekstowych dane będą zawsze pobierane z historii.

Funkcja agregująca	Określ, której funkcji agregującej użyć w wybranym *Okresie*: **min** - wyświetl najmniejszą wartość; **max** - wyświetl największą wartość; **avg** - wyświetl wartość średnią; **count** - wyświetl liczbę wartości; **sum** - wyświetl sumę wartości; **first** - wyświetl pierwszą wartość; **last** - wyświetl ostatnią wartość. Jeśli wybrano **nie używaj**, wyświetlana jest najnowsza wartość (bez agregacji). Agregacja umożliwia wyświetlenie zagregowanej wartości dla określonego przedziału (5 minut, godziny, dnia) zamiast najnowszej wartości. Dla funkcji *min*, *max*, *avg* i *sum* można wyświetlać tylko dane liczbowe. W przypadku funkcji *count* dane nieliczbowe zostaną zmienione na liczbowe.
Okres	Określ okres używany do agregowania wartości. Wybierz źródło danych dla okresu: **Dashboard** - użyj selektora okresu dashboardu; **Widget** - użyj kompatybilnego widgetu (ustawionego w parametrze *Widget*); **Niestandardowy** - użyj niestandardowego okresu ustawionego w parametrach *Od* i *Do*; jeśli jest ustawiony, w prawym górnym rogu widgetu zostanie wyświetlona ikona zegara wskazująca ustawiony czas po najechaniu kursorem myszy. Należy pamiętać, że niezależnie od konfiguracji parametru *Okres* widgetu kompatybilne widgety nadal mogą używać go jako źródła danych dla okresu. Ten parametr nie jest dostępny, jeśli dla *Funkcji agregującej* ustawiono „nie używaj”.
Widget	Wprowadź lub wybierz kompatybilny widget jako źródło danych dla okresu. Ten parametr jest dostępny, jeśli dla *Okresu* ustawiono „Widget”.
Od	Wprowadź lub wybierz początek okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Okresu* ustawiono „Niestandardowy”.
Do	Wprowadź lub wybierz koniec okresu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli dla *Okresu* ustawiono „Niestandardowy”.
Agreguj kolumny	Po zaznaczeniu wszystkie pozycje pasujące do wzorców pozycji kolumny zostaną zgrupowane w jednej połączonej kolumnie/wierszu. Wartości pasujących pozycji zostaną zagregowane przy użyciu wybranej *Funkcji agregującej kolumny* i wyświetlone pod *Nazwą połączonej kolumny*. Domyślnie: wyłączone (niezaznaczone).Ten parametr jest wyświetlany tylko wtedy, gdy *Wyświetlaj wartość jako* ma ustawioną wartość **Liczbowe**.
Funkcja agregująca kolumny	Wybierz sposób łączenia wartości z pogrupowanych wzorców pozycji. Dostępne opcje: **min**, **max**, **avg**, **count**, **sum** (domyślnie). Ten parametr jest wyświetlany tylko wtedy, gdy zaznaczono *Agreguj kolumny*.
Nazwa połączonej kolumny	Wprowadź nazwę wyświetlaną dla zagregowanej kolumny. To pole jest wymagane, gdy zaznaczono *Agreguj kolumny*, i staje się nagłówkiem kolumny dla połączonych wartości.

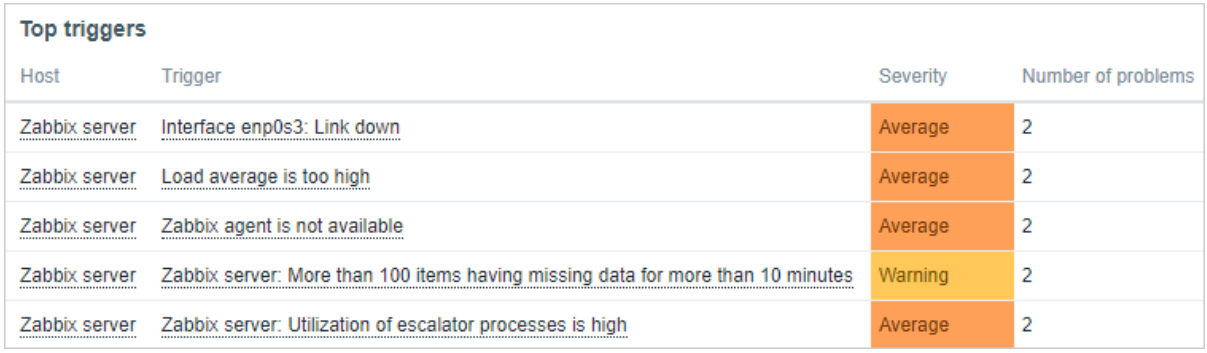
Parametry kolumn dla wartości tekstowych:

Wyróżnienia	Określ wyrażenia regularne, po dopasowaniu do których powinien zmienić się kolor tła/wypełnienia.

Najważniejsze wyzwalacze

Przegląd

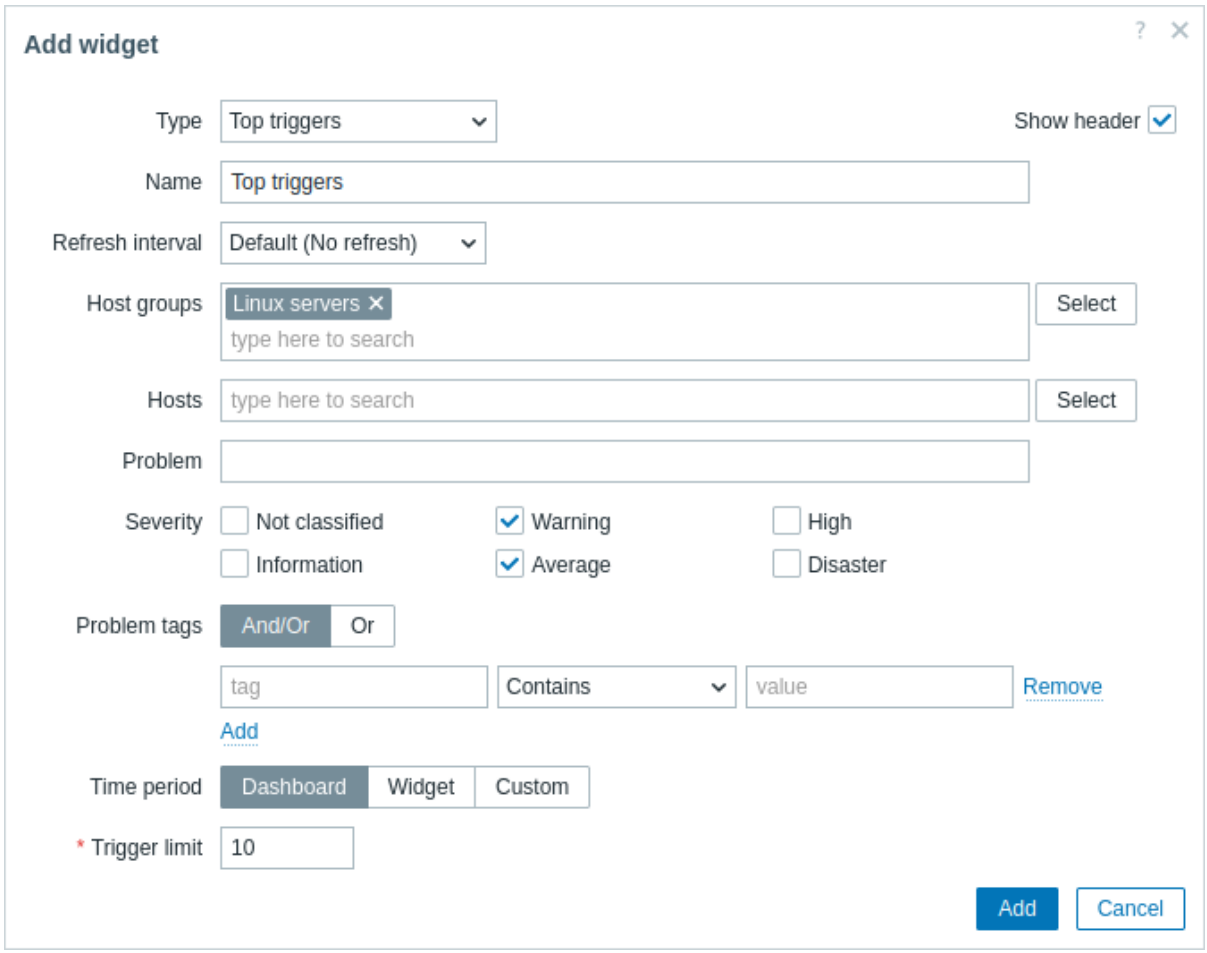
Widżet *Top triggers* wyświetla wyzwalacze z największą liczbą problemów.



Wyświetla te same dane co *Raporty > Top 100 triggers* i może wyświetlić do 1000 rekordów.

Konfiguracja

Aby skonfigurować widget, wybierz *Top triggers* jako typ widgetu:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

Host groups	Wybierz grupy hostów, których problemy mają być wyświetlane w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Problemy z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną wprowadzone żadne grupy hostów, wyświetlane będą problemy ze wszystkich grup hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na dashboardzie szablonu.
Hosts	Wybierz hosty, których problemy mają być wyświetlane w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli nie zostaną wprowadzone żadne hosty, wyświetlane będą problemy wszystkich hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na dashboardzie szablonu.

Problem	Możesz wyświetlać wyzwalacze tylko dla określonych problemów. W tym celu wprowadź ciąg znaków, który ma zostać dopasowany w nazwie problemu. Makra nie są rozwijane.
Severity	Zaznacz poziomy ważności wyzwalaczy, aby filtrować wyzwalacze wyświetlane w widżecie. Jeśli nie zostanie zaznaczony żaden poziom ważności, wyświetlane będą wszystkie wyzwalacze.
Problem tags	Określ tagi problemów, które mają być wyświetlane w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane warunkiem *Or*; **Or** - wystarczy spełnienie jednego warunku.
Time period	Wybierz źródło danych dla okresu czasu: **Dashboard** - użyj selektora okresu czasu dashboardu; **Widget** - użyj kompatybilnego widgetu (ustawionego w parametrze *Widget*); **Custom** - użyj niestandardowego okresu czasu ustawionego w parametrach *From* i *To*; jeśli zostanie ustawiony, w prawym górnym rogu widgetu będzie wyświetlana ikona zegara, wskazująca ustawiony czas po najechaniu kursorem. Zwróć uwagę, że niezależnie od konfiguracji *Time period* widgetu, kompatybilne widgety nadal mogą używać go jako źródła danych dla okresu czasu.
Widget	Wprowadź lub wybierz kompatybilny widget jako źródło danych dla okresu czasu. Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Widget".
From	Wprowadź lub wybierz początek okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Custom".
To	Wprowadź lub wybierz koniec okresu czasu. Obsługiwana jest składnia czasu względnego (`now`, `now/d`, `now/w-1w` itd.). Ten parametr jest dostępny, jeśli *Time period* jest ustawiony na "Custom".
Trigger limit	Ustaw liczbę wyzwalaczy do wyświetlenia. Możliwy zakres wartości: 1-1000.

Przegląd wyzwalaczy

Przegląd

Widżet *Trigger overview* wyświetla bieżące stany wyzwalaczy hostów w formie tabeli kolorowych bloków. Zapewnia szybki wizualny podgląd kondycji i aktywności wyzwalaczy na różnych hostach.



Widżet używa kilku elementów wizualnych do przedstawienia szczegółów wyzwalaczy:

- Kolory bloków dla wyzwalaczy PROBLEM odpowiadają poziomom ważności problemów.
- Migające bloki oznaczają ostatnie zmiany stanu wyzwalacza (w ciągu ostatnich 2 minut).
- Szare strzałki oznaczają wyzwalacze z zależnościami (najedź na nie kursorem, aby zobaczyć szczegóły).
- Ikony pól wyboru oznaczają potwierdzone problemy; aby ta ikona była widoczna, wszystkie aktywne lub rozwiązane problemy dla wyzwalacza muszą być potwierdzone.

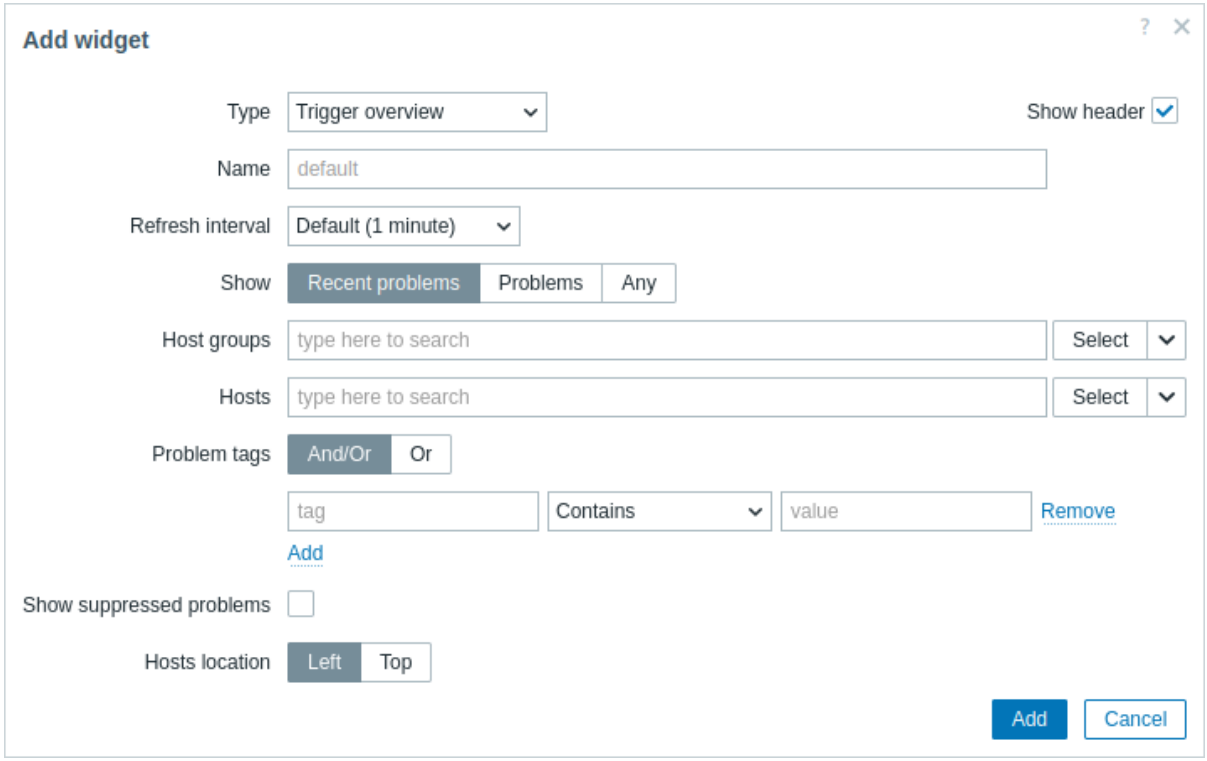
Note:
Kolory ważności wyzwalaczy można dostosować w *Administration* > *General*. Ważności wyzwalaczy można modyfikować podczas aktualizowania problemów.

Kliknięcie bloku udostępnia łącza do informacji związanych z wyzwalaczem (problemy, konfiguracja wyzwalacza itp.).

Domyślnie wyświetlanych jest do 50 rekordów (można to skonfigurować w *Administration* > *General* > *GUI* w sekcji *Maksymalna liczba kolumn i wierszy w tabelach przeglądu*). Jeśli istnieje więcej wyzwalaczy, pojawi się komunikat z prośbą o zawężenie kryteriów filtrowania. To ograniczenie jest stosowane przed wszelkim dodatkowym filtrowaniem (na przykład według tagów).

Konfiguracja

Aby skonfigurować widget, wybierz *Przegląd wyzwalaczy* jako typ:



Oprócz parametrów wspólnych dla wszystkich widgetów, możesz ustawić następujące opcje specyficzne:

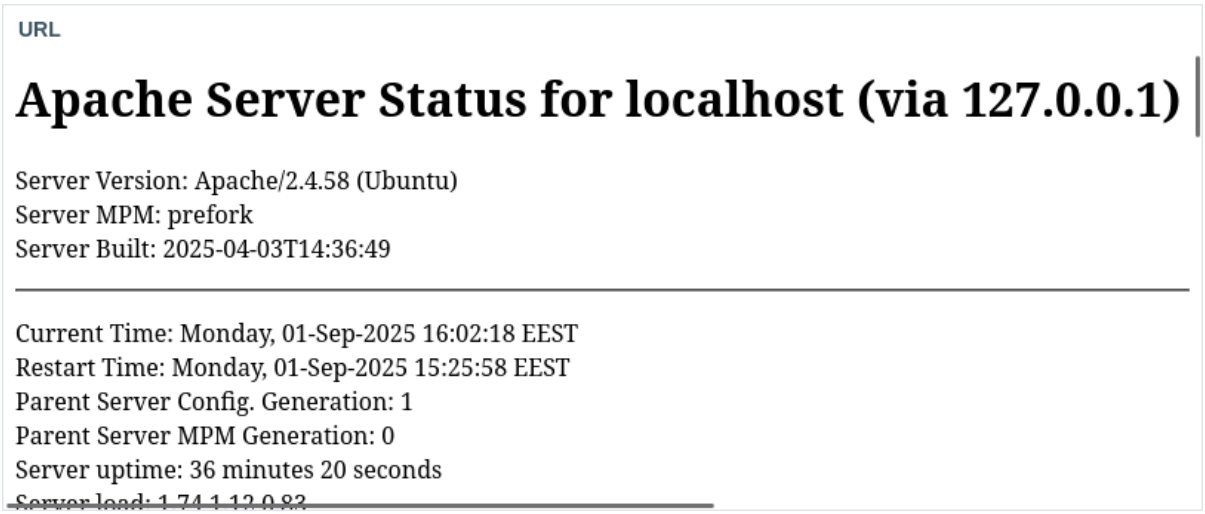
Pokaż	Filtruj wyzwalacze według stanu wyzwalacza: **Ostatnie problemy** - *(domyślnie)* pokazuj wyzwalacze, które niedawno były lub nadal są w stanie PROBLEM (rozwiązane i nierozwiązane); **Problemy** - pokazuj wyzwalacze, które są w stanie PROBLEM (nierozwiązane); **Dowolne** - pokazuj wszystkie wyzwalacze.
Grupy hostów	Wybierz grupy hostów. Alternatywnie wybierz zgodny widget jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.
Hosty	Wybierz hosty. Alternatywnie wybierz zgodny widget lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Ten parametr nie jest dostępny podczas konfigurowania widgetu na pulpicie szablonu.

Tagi problemów	Określ tagi do filtrowania wyzwalaczy wyświetlanych w widżecie. Można zarówno uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. **Uwaga:** Jeśli parametr *Pokaż* jest ustawiony na „Dowolne”, zostaną wyświetlone wszystkie wyzwalacze, nawet jeśli określono tagi. Jednak chociaż ostatnie zmiany stanu wyzwalacza (wyświetlane jako migające bloki) będą aktualizowane dla wszystkich wyzwalaczy, szczegóły stanu wyzwalacza (kolor ważności problemu oraz informacja, czy problem został potwierdzony) będą aktualizowane tylko dla wyzwalaczy pasujących do określonych tagów. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, dla których wartości tagów zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, a warunki mające tę samą nazwę tagu będą grupowane przez warunek *Or*; **Or** - wystarczy spełnienie jednego warunku.
Pokaż ukryte problemy	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby ukryte (niepokazywane) z powodu konserwacji hosta.
Układ	Wybierz opcję układu: **Poziomy** - nazwy hostów będą wyświetlane po lewej stronie; **Pionowy** - nazwy hostów będą wyświetlane u góry.

URL

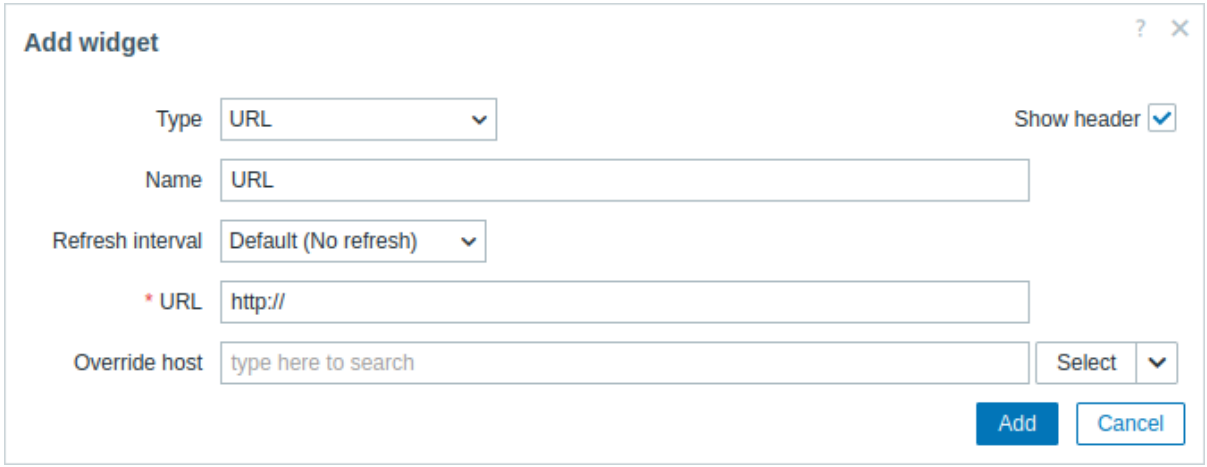
Przegląd

Widżet *URL* wyświetla zawartość pobraną z adresu URL.



Konfiguracja

Aby skonfigurować, wybierz *URL* jako typ:



Oprócz parametrów wspólnych dla wszystkich widżetów możesz ustawić następujące opcje specyficzne:

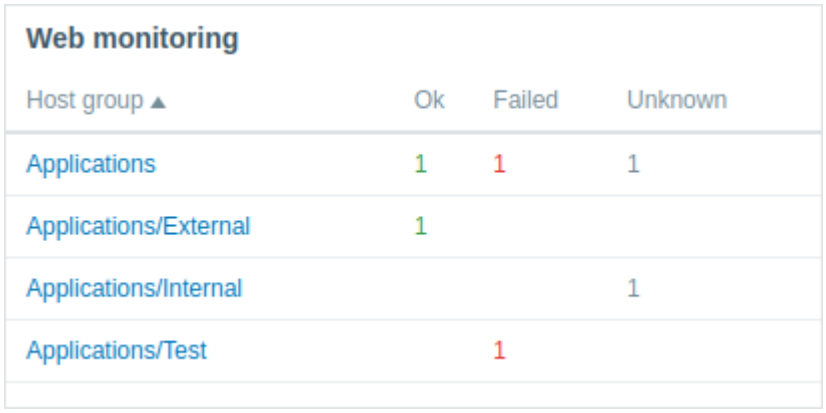
URL	Wprowadź URL do wyświetlenia (do 2048 znaków). Zewnętrzne adresy URL muszą zaczynać się od `http://` lub `https://`. Wewnętrzne adresy URL obsługują ścieżki względne (na przykład `zabbix.php?action=report.status`). Obsługiwane są makra `{HOST.*}`.
Override host	Wybierz zgodny widżet lub selektor hosta host selector na pulpicie jako źródło danych dla hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na szablonowym pulpicie.

Attention:
Przeglądarki mogą nie wczytać strony HTTP skonfigurowanej w widżecie, jeśli frontend Zabbixa jest dostępny przez HTTPS. Może być również konieczna modyfikacja nagłówka odpowiedzi HTTP X-Frame-Options, aby wyświetlić zawartość zewnętrzną względem frontend Zabbixa.

Monitorowanie sieci web

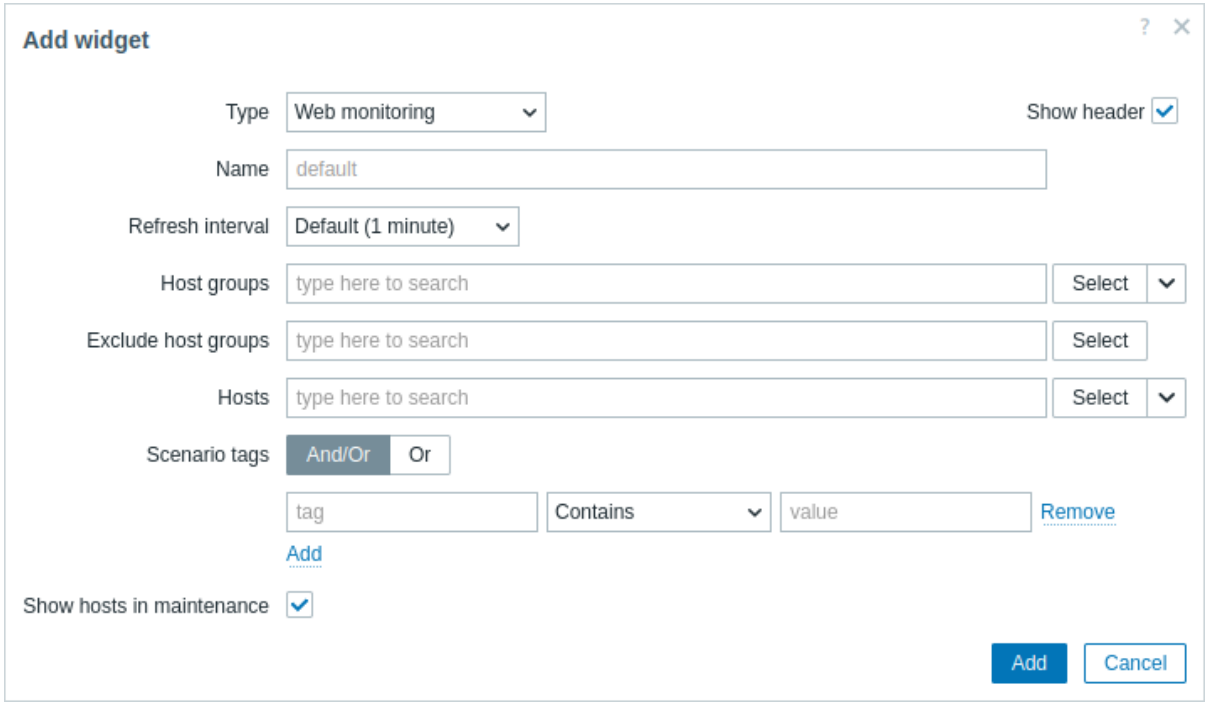
Przegląd

Widżet *Monitorowanie WWW* wyświetla stan aktywnych scenariuszy WWW.



Kliknięcie nazwy grupy hostów otwiera *Monitorowanie* > *Hosty* > *Scenariusze WWW*, gdzie można wyświetlić szczegóły scenariusza.

Konfiguracja



Oprócz parametrów wspólnych dla wszystkich widżetów można ustawić następujące opcje specyficzne:

Grupy hostów	Wybierz grupy hostów do wyświetlenia w widżecie. Alternatywnie wybierz zgodny widżet jako źródło danych dla grup hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów będą wyświetlane w widżecie; jeśli nie zostaną wprowadzone żadne grupy hostów, wyświetlone zostaną wszystkie grupy hostów. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Wyklucz grupy hostów	Wybierz grupy hostów, które mają zostać ukryte w widżecie. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy grupy zostanie wyświetlona rozwijana lista pasujących grup. Określenie nadrzędnej grupy hostów powoduje automatyczne wybranie wszystkich zagnieżdżonych grup hostów. Dane hostów z tych grup hostów nie będą wyświetlane w widżecie. Na przykład hosty 001, 002, 003 mogą należeć do Grupy A, a hosty 002, 003 również do Grupy B. Jeśli jednocześnie wybierzemy *pokazywanie* Grupy A i *wykluczenie* Grupy B, na pulpicie zostaną wyświetlone tylko dane hosta 001. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	Wybierz hosty do wyświetlenia w widżecie. Alternatywnie wybierz zgodny widżet lub pulpit jako źródło danych dla hostów. To pole obsługuje autouzupełnianie, więc po rozpoczęciu wpisywania nazwy hosta zostanie wyświetlona rozwijana lista pasujących hostów. Jeśli nie zostaną wprowadzone żadne hosty, wyświetlone zostaną wszystkie hosty. Ten parametr nie jest dostępny podczas konfigurowania widżetu na pulpicie szablonu.

Tagi scenariuszy	Określ tagi, aby ograniczyć liczbę scenariuszy WWW wyświetlanych w widżecie. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy tagów i wartości (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez uwzględniania wielkości liter). Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane warunkiem *Or*; **Or** - wystarczy, że jeden warunek zostanie spełniony.
Pokaż hosty w konserwacji	Uwzględnij w statystykach hosty będące w konserwacji. Ten parametr ma etykietę *Pokaż dane w konserwacji* podczas konfigurowania widżetu na pulpicie szablonu.

Monitorowanie

Przegląd

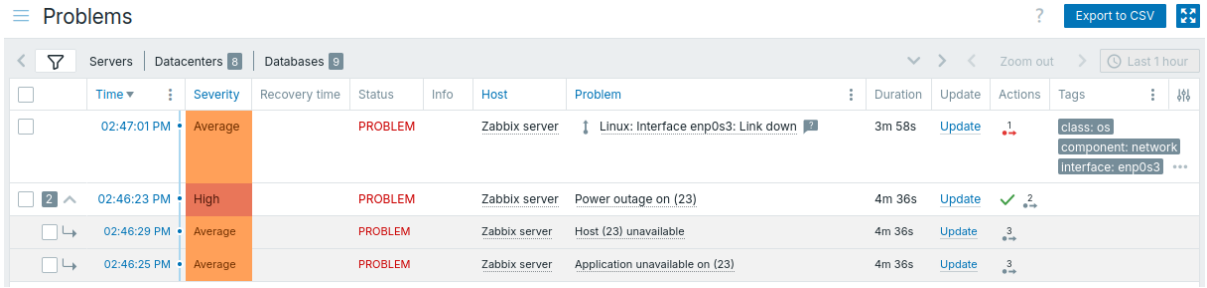
Menu Monitorowanie służy do wyświetlania danych. Wszystkie informacje, które Zabbix został skonfigurowany do gromadzenia, wizualizacji i przetwarzania, będą wyświetlane w różnych sekcjach menu *Monitorowanie*.

Problemy

Omówienie

W *Monitoring* > *Problems* możesz zobaczyć, jakie problemy masz obecnie. Problemy to wyzwalacze, które znajdują się w stanie "Problem".

Domyślnie wszystkie nowe problemy są klasyfikowane jako problemy przyczynowe. Możliwe jest ręczne przeklasyfikowanie niektórych problemów jako problemy objawowe problemu przyczynowego. Więcej informacji znajdziesz w sekcji zdarzenia przyczynowe i objawowe.

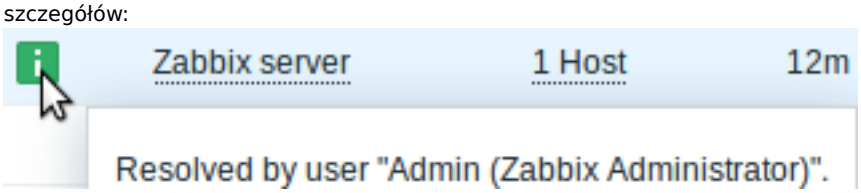
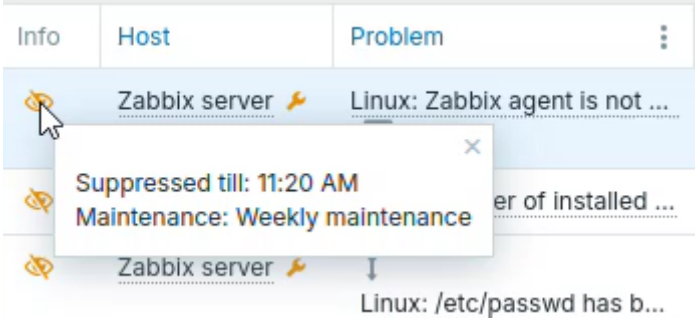


Szczegóły problemu są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia pokazywanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie układu domyślnego. Dostępne są również następujące opcje:

- *Widok kompaktowy* - zmniejsza wysokość wierszy, aby na ekranie zmieściło się więcej problemów.
- *Podświetl cały wiersz* - używa koloru ważności problemu do podświetlania nierozwiązanych problemów (niedostępne w motywach o wysokim kontraście).

Możesz też zmieniać rozmiar każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślny rozmiar. Wszystkie dostosowania są stosowane natychmiast i są zawsze zapamiętywane dla bieżącego użytkownika w aktywnej karcie filtra (domyślnej lub zapisanej).

Dostępne są następujące kolumny:

Column	Description
Checkbox	Wyświetlane są pola wyboru do zaznaczania problemów. Ikony obok pól wyboru mają następujące znaczenie:  - liczba zdarzeń objawowych dla problemu przyczynowego;  - rozwiń, aby pokazać zdarzenia objawowe;  - zwiń, aby ukryć zdarzenia objawowe;  - to jest zdarzenie objawowe.
Time	Wyświetlany jest czas rozpoczęcia problemu. Kliknięcie przycisku *Opcje* w nagłówku kolumny daje dostęp do opcji *Pokaż oś czasu*, która przełącza wizualną oś czasu i grupowanie problemów (niedostępne, gdy włączony jest *Widok kompaktowy*).
Severity	Wyświetlana jest ważność problemu. Ważność problemu jest pierwotnie oparta na ważności podstawowego wyzwalacza problemu, jednak po wystąpieniu zdarzenia może zostać zaktualizowana za pomocą ekranu *Aktualizuj problem* screen. Kolor ważności problemu jest używany jako tło komórki w czasie trwania problemu.
Recovery time	Wyświetlany jest czas rozwiązania problemu Ta kolumna nie jest dostępna, gdy opcja *Pokaż* w filtrze jest ustawiona na *Problems*.
Status	Wyświetlany jest status problemu: **Problem** - nierozwiązany problem **Resolved** - niedawno rozwiązany problem. Możesz ukryć niedawno rozwiązane problemy za pomocą filtra. Nowe i niedawno rozwiązane problemy migają przez 2 minuty. Rozwiązane problemy są wyświetlane łącznie przez 5 minut. Obie te wartości można skonfigurować w *Administration* > *General* > *Trigger displaying options*. Ta kolumna nie jest dostępna, gdy opcja *Pokaż* w filtrze jest ustawiona na *Problems*.
Info	Wyświetlana jest zielona ikona informacji, jeśli problem został zamknięty przez globalną korelację lub ręcznie podczas aktualizacji problemu. Najechanie kursorem na ikonę wyświetla więcej szczegółów:  Jeśli wyświetlany jest problem wstrzymany, pokazana jest następująca ikona:  . Gdy zdarzenie jest wstrzymane przez konserwację hosta, ikona jest wyświetlana na pomarańczowo:  . Jeśli obowiązuje zarówno konserwacja, jak i ręczne wstrzymanie, ręczne wstrzymanie ma pierwszeństwo w kolorze ikony. Najechanie kursorem na ikonę wyświetla więcej szczegółów: 
Host	Wyświetlany jest host problemu. Kliknięcie nazwy hosta otwiera menu hosta.

Column	Description
Problem	Wyświetlana jest nazwa problemu. Nazwa problemu jest oparta na nazwie podstawowego wyzwalacza problemu. Makra w nazwie wyzwalacza są rozwijane w momencie wystąpienia problemu, a rozwinięte wartości nie są już aktualizowane. Kliknięcie nazwy problemu otwiera menu zdarzenia. Najechanie na ikonę  po nazwie problemu wyświetli opis wyzwalacza (dla tych problemów, które go mają). Kliknięcie przycisku *Opcje* w nagłówku kolumny daje dostęp do dwóch opcji: - *Pokaż dane operacyjne* - wyświetla dane operacyjne po nazwie problemu w nawiasach (tylko jeśli pole *Operational data* nie jest puste w konfiguracji wyzwalacza). - *Pokaż wyrażenie wyzwalacza* - wyświetla wyrażenie wyzwalacza problemu pod nazwą problemu (niedostępne, gdy włączony jest *Widok kompaktowy*).
Operational data	Wyświetlane są dynamiczne dane w czasie rzeczywistym dotyczące problemu. Zawartość zależy od pola *Operational data* w konfiguracji wyzwalacza: jeśli zdefiniowany jest ciąg z makrami, zostanie on tutaj pokazany; jeśli pole pozostanie puste, zamiast tego wyświetlone zostaną najnowsze wartości wszystkich pozycji w wyrażeniu wyzwalacza.
Duration	Wyświetlany jest czas trwania problemu. Zobacz też: Ujemny czas trwania problemu
Update	Kliknij link *Update*, aby przejść do ekranu aktualizacji problemu, gdzie można wykonać różne działania związane z problemem, w tym dodać komentarz i potwierdzić problem.
Actions	Historia działań dotyczących problemu jest wyświetlana za pomocą symbolicznych ikon:  - problem został potwierdzony. Ta ikona jest zawsze wyświetlana jako pierwsza.  - dodano komentarze. Wyświetlana jest również liczba komentarzy.  - ważność problemu została zwiększona (np. Information > Warning).  - ważność problemu została zmniejszona (np. Warning > Information).  - ważność problemu została zmieniona, ale wróciła do pierwotnego poziomu (np. Warning > Information > Warning).  - problem został wstrzymany.  - wstrzymanie problemu zostało usunięte.  - problem został wstrzymany z powodu konserwacji hosta.  - wstrzymanie problemu zostało usunięte po konserwacji hosta.  - wykonano działania. Wyświetlana jest również liczba działań.  - wykonano działania, co najmniej jedno jest w toku. Wyświetlana jest również liczba działań.  - wykonano działania, co najmniej jedno zakończyło się niepowodzeniem. Wyświetlana jest również liczba działań. Po najechaniu myszą na ikony wyświetlane są okna podręczne ze szczegółami dotyczącymi aktywności. Zobacz wyświetlanie szczegółów, aby dowiedzieć się więcej o ikonach używanych w oknie podręcznym dla wykonanych działań.
Tags	Wyświetlane są tagi problemu. Ponadto mogą być również wyświetlane tagi z zewnętrznego systemu zgłoszeń (zobacz opcję *Process tags* podczas konfigurowania webhooków). Najechanie kursorem lub kliknięcie ikony trzech kropek po wyświetlonych tagach otwiera okno podręczne ze wszystkimi tagami. Kliknięcie przycisku *Opcje* w nagłówku kolumny pozwala ustawić liczbę wyświetlanych tagów (1-3), wybrać sposób wyświetlania nazw tagów (pełne, skrócone lub brak), ustawić priorytet wyświetlania tagów lub zduplikować kolumnę (zduplikowane kolumny można zmieniać nazwę lub usuwać).

Column	Description
Tag value	Wyświetlane są wartości tagów problemu. Kliknięcie przycisku ⋮ *Opcje* w nagłówku kolumny pozwala zmienić nazwę kolumny, wybrać, które wartości tagów mają być wyświetlane, lub zduplikować kolumnę (zduplikowane kolumny można zmieniać nazwę lub usuwać).
Custom text	Wyświetlany jest tekst niestandardowy, wartości makr lub ich kombinacja. Kliknięcie przycisku ⋮ *Opcje* w nagłówku kolumny pozwala zmienić nazwę kolumny, wprowadzić tekst niestandardowy lub makra, albo zduplikować kolumnę (zduplikowane kolumny można zmieniać nazwę lub usuwać). Obsługiwane makra: `{HOST.*}` (z wyjątkiem `{HOST.METADATA}`), `{INVENTORY.*}`, `{ITEM.*}` (z wyjątkiem `{ITEM.STATE.ERROR}`), `{TRIGGER.ID}`, `{EVENT.ID}` oraz makra użytkownika.

Ujemny czas trwania problemu

W niektórych typowych sytuacjach rzeczywiście możliwe jest wystąpienie ujemnego czasu trwania problemu, tj. gdy czas rozwiązania problemu jest wcześniejszy niż czas utworzenia problemu, np.:

- Jeśli jakiś host jest monitorowany przez proxy i wystąpi błąd sieci, w wyniku którego przez pewien czas nie są odbierane dane z proxy, wyzwalacz nodata(/host/key) zostanie uruchomiony przez serwer. Po przywróceniu połączenia serwer otrzyma dane pozycji z proxy z czasem z przeszłości. Następnie problem nodata(/host/key) zostanie rozwiązany i będzie miał ujemny czas trwania problemu;
- Gdy dane pozycji rozwiązujące zdarzenie problemu są wysyłane przez Zabbix sender i zawierają znacznik czasu wcześniejszy niż czas utworzenia problemu, również zostanie wyświetlony ujemny czas trwania problemu.

Note:
Ujemny czas trwania problemu nie wpływa w żaden sposób na obliczanie SLA ani raport dostępności dla danego wyzwalacza; nie skraca ani nie wydłuża czasu trwania problemu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Masowa aktualizacja* - aktualizuje wybrane problemy, przechodząc do ekranu aktualizacji problemu

Aby użyć tej opcji, zaznacz pola wyboru przed odpowiednimi problemami, a następnie kliknij przycisk *Masowa aktualizacja*.

Przyciski

Przycisk po prawej stronie oferuje następującą opcję:

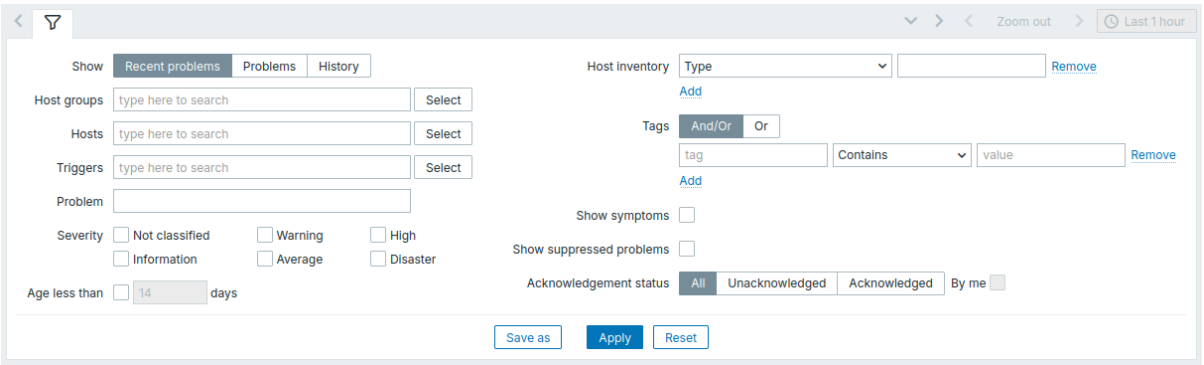
Export to CSV	Eksportuj zawartość ze wszystkich stron do pliku CSV.

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko problemy, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Filtr znajduje się nad tabelą. Ulubione ustawienia filtra można zapisać jako karty, a następnie szybko otwierać, klikając karty nad filtrem.

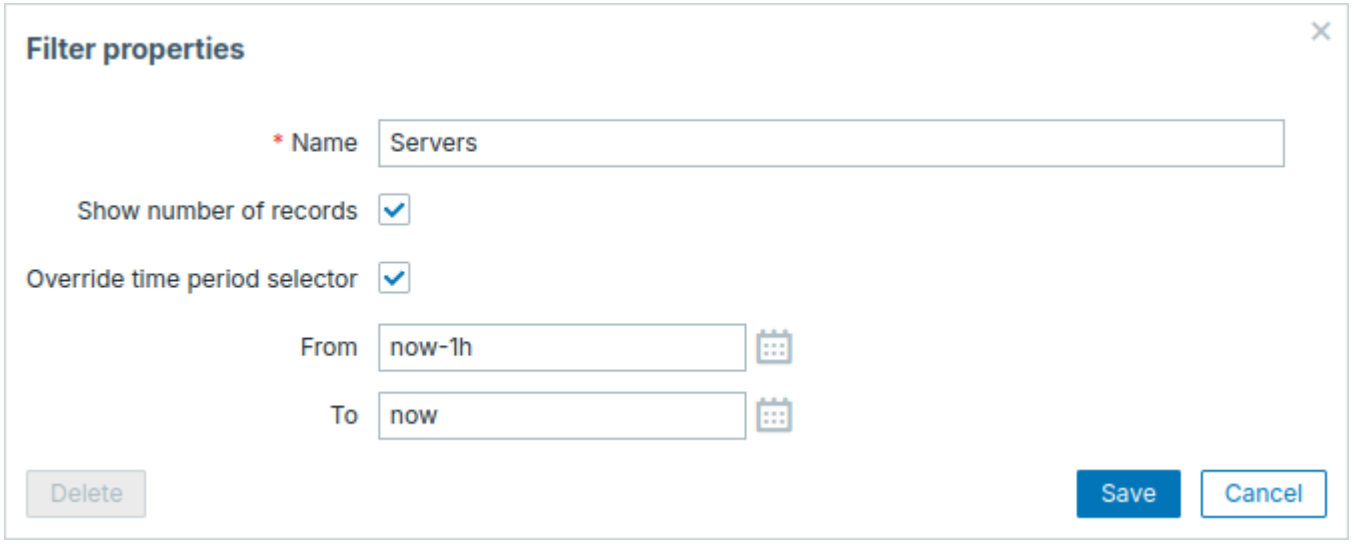


Parameter	Description
Show	Filtruj według statusu problemu: **Recent problems** - wyświetlane są nierozwiązane i niedawno rozwiązane problemy (domyślnie) **Problems** - wyświetlane są nierozwiązane problemy **History** - wyświetlana jest historia wszystkich zdarzeń
Host groups	Filtruj według jednej lub kilku grup hostów. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów.
Hosts	Filtruj według jednego lub kilku hostów.
Triggers	Filtruj według jednego lub kilku wyzwalaczy.
Problem	Filtruj według nazwy problemu.
Severity	Filtruj według powagi wyzwalacza (problemu).
Age less than	Filtruj według wieku problemu.
Host inventory	Filtruj według typu i wartości inwentarza hosta.
Tags	Filtruj według nazwy i wartości tagu zdarzenia. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony jest jeden warunek Po zastosowaniu filtra tagi określone tutaj będą wyświetlane jako pierwsze przy problemie, chyba że zostanie to zastąpione przez listę *Tag display priority* (patrz poniżej).
Show symptoms	Zaznacz pole wyboru, aby wyświetlać na osobnej linii problemy sklasyfikowane jako objawy.
Show suppressed problems	Zaznacz pole wyboru, aby wyświetlać problemy, które w przeciwnym razie byłyby tłumione i ukryte z powodu konserwacji hosta lub tłumione ręcznie przez użytkownika.
Acknowledgement status	Filtr umożliwiający wyświetlenie wszystkich problemów, tylko nierozpoznanych problemów lub tylko rozpoznanych problemów. Zaznacz dodatkowe pole wyboru, aby odfiltrować problemy, które kiedykolwiek zostały przez Ciebie rozpoznane.

Karty ulubionych filtrów

Często używane zestawy parametrów filtra można zapisywać w kartach.

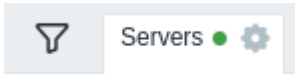
Aby zapisać nowy zestaw parametrów filtra, otwórz główną kartę, skonfiguruj ustawienia filtra, a następnie kliknij przycisk *Save as*. W nowym oknie podręcznym zdefiniuj *Filter properties*.



Parameter	Description
Name	Nazwa filtra wyświetlana na liście kart.
Show number of records	Zaznacz, jeśli chcesz, aby liczba problemów była wyświetlana obok nazwy karty.
Override time period selector	Zaznacz, aby ustawić określony domyślny okres czasu dla tego zestawu filtrów. Jeśli ta opcja jest ustawiona, zmianę okresu czasu dla tej karty będzie można wykonać tylko przez aktualizację ustawień filtra. W przypadku kart bez niestandardowego okresu czasu zakres czasu można zmienić, klikając przycisk selektora czasu w prawym górnym rogu (nazwa przycisku zależy od wybranego przedziału czasu: This week, Last 30 minutes, Yesterday itp.). Ta opcja jest dostępna tylko dla filtrów w *Monitoring* > *Problems*.
From/To	Początek i koniec okresu czasu w składni czasu bezwzględnego (`Y-m-d H:i:s`) lub względnego (`now-1d`). Dostępne, jeśli zaznaczono *Set custom time period*.

Po zapisaniu filtr zostaje utworzony jako nazwany filtr w karcie i natychmiast aktywowany.

Aby edytować właściwości filtra istniejącego filtra, kliknij symbol koła zębatego obok nazwy aktywnej karty.



Uwagi:

- Aby ukryć obszar filtra, kliknij nazwę bieżącej karty. Kliknij ponownie nazwę aktywnej karty, aby ponownie otworzyć obszar filtra.
- Obsługiwana jest nawigacja klawiaturą: użyj strzałek, aby przełączać się między kartami, naciśnij *Enter*, aby otworzyć.
- Przyciski lewo/prawo nad filtrem mogą służyć do przełączania między zapisanymi filtrami. Alternatywnie przycisk skierowany w dół otwiera menu rozwijane ze wszystkimi zapisanymi filtrami i można kliknąć ten, którego potrzebujesz.
- Jeśli ustawienia zapisanego filtra zostały zmienione (ale nie zapisane), po nazwie filtra wyświetlana jest zielona kropka. Aby zaktualizować filtr zgodnie z nowymi ustawieniami, kliknij przycisk *Update*, który jest wyświetlany zamiast przycisku *Save as*.
- Układ kolumn tabeli jest zapisywany osobno dla każdej karty filtra. Aby dostosować układ zapisanego filtra, najpierw otwórz tę kartę filtra, a następnie dostosuj kolumny.
- Karty filtrów, ustawienia filtrów oraz układ kolumn tabeli dla każdej karty są zapamiętywane w profilu użytkownika i przywracane, gdy użytkownik ponownie otworzy stronę.

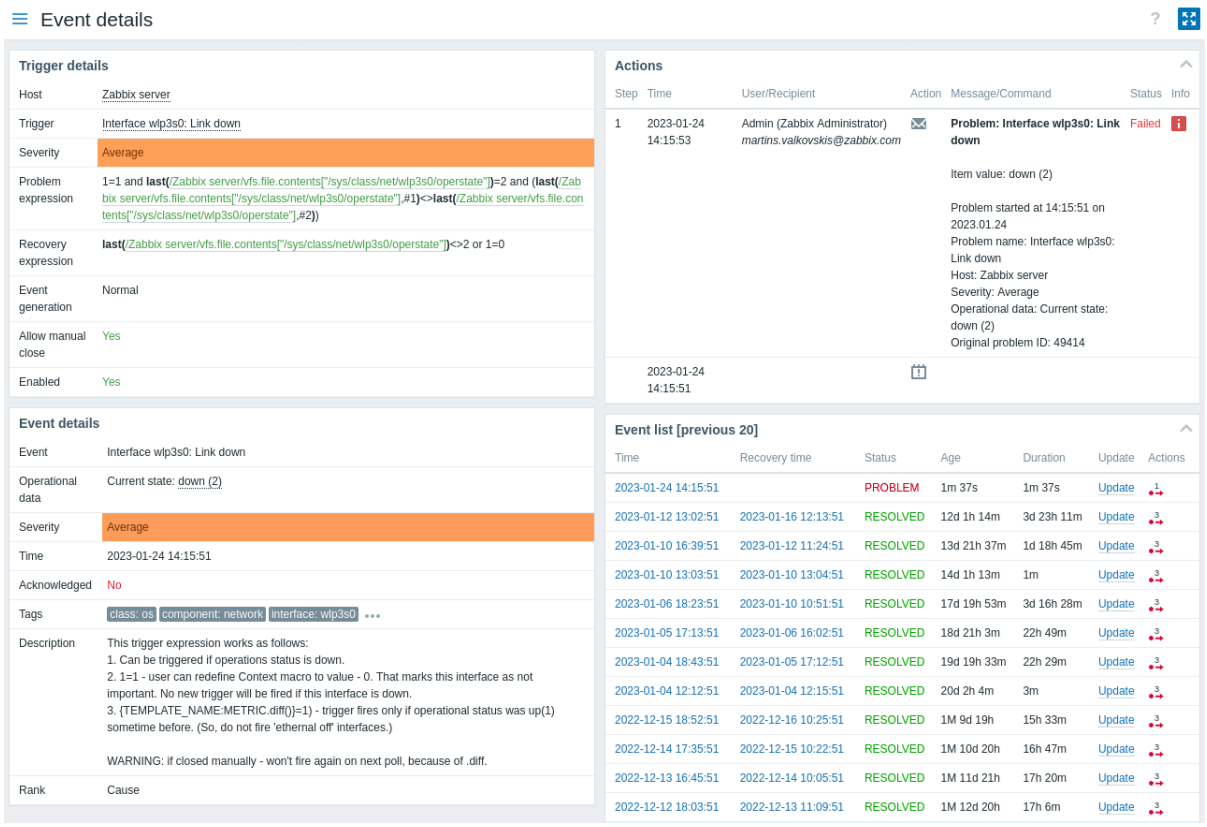
Note:
Aby udostępniać filtry, skopiuj i wyślij innym adres URL aktywnego filtra. Po otwarciu tego adresu URL inni użytkownicy będą mogli zapisać ten zestaw parametrów jako trwały filtr na swoim koncie Zabbix.
 Zobacz też: Page parameters.

Przyciski filtra

Apply	Zastosuj określone kryteria filtrowania (bez zapisywania).
Reset	Zresetuj bieżący filtr i przywróć zapisane parametry bieżącej karty. Na karcie głównej spowoduje to wyczyszczenie filtra.
Save as	Zapisz bieżące parametry filtra na nowej karcie. Dostępne tylko na karcie głównej.
Update	Zastąp parametry karty aktualnie określonymi parametrami. Niedostępne na karcie głównej.

Wyświetlanie szczegółów

Czasy rozpoczęcia problemu i odzyskania w *Monitoring* > *Problems* są linkami. Kliknięcie ich otwiera więcej szczegółów zdarzenia.



W szczegółach zdarzenia można zobaczyć, czy zdarzenie zostało wstrzymane z powodu konserwacji lub ręcznie przez użytkownika.

Należy pamiętać, że poziom istotności problemu może różnić się dla wyzwalacza i zdarzenia problemu - jeśli został zaktualizowany dla zdarzenia problemu za pomocą ekranu *Update problem* screen.

Na liście działań używane są następujące ikony do oznaczania typu aktywności:

- - wygenerowano zdarzenie problemu.
- - wysłano wiadomość.
- - zdarzenie problemu potwierdzone.
- - zdarzenie problemu niepotwierdzone.
- - dodano komentarz.
- - poziom istotności problemu został podwyższony (np. Information > Warning).
- - poziom istotności problemu został obniżony (np. Warning > Information).
- - poziom istotności problemu został zmieniony, ale wrócił do pierwotnego poziomu (np. Warning > Information > Warning).
- - wykonano zdalne polecenie.
- - zdarzenie problemu zostało odzyskane.
- - problem został zamknięty ręcznie.
- - problem został wstrzymany.
- - wstrzymanie problemu zostało usunięte.
- - problem został wstrzymany z powodu konserwacji hosta.
- - wstrzymanie problemu zostało usunięte po konserwacji hosta.
- - problem został przekształcony w problem objawowy.
- - problem został przekształcony w problem przyczynowy.

Przyczyny i objawy problemów

Przegląd

Domyślnie wszystkie nowe problemy są klasyfikowane jako problemy przyczynowe. Możliwe jest ręczne przeklasyfikowanie określonych problemów na problemy objawowe problemu przyczynowego.

Na przykład awaria zasilania może być rzeczywistą przyczyną źródłową, przez którą jakiś host jest nieosiągalny lub jakaś usługa nie działa. W takim przypadku problemy „host jest nieosiągalny" i „usługa nie działa" muszą zostać sklasyfikowane jako problemy objawowe problemu „awaria zasilania" — problemu przyczynowego.

Hierarchia przyczyna-objaw obsługuje tylko dwa poziomy. Problemu, który jest już objawem, nie można przypisać „podrzędnych" problemów objawowych; wszelkie problemy przypisane jako objawy do problemu objawowego staną się objawami tego samego problemu przyczynowego.

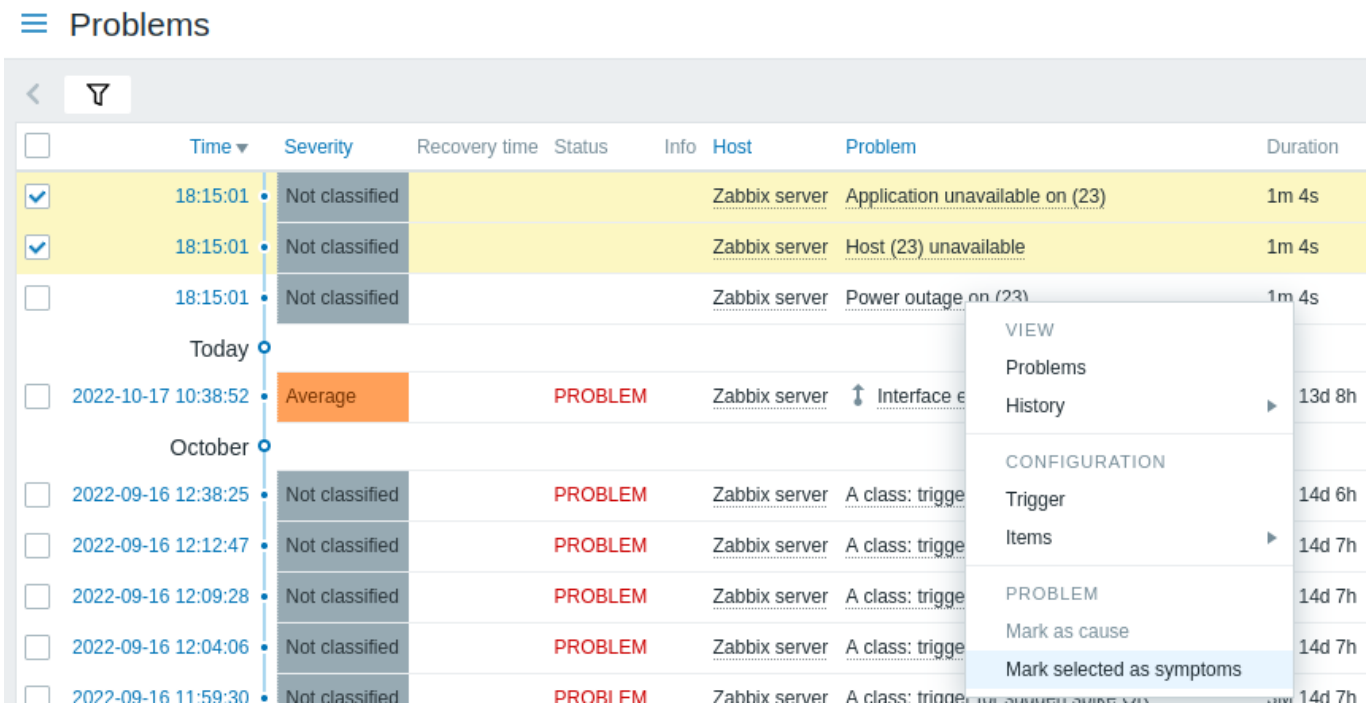
W sumach problemów na mapach, w widżetach pulpitu, takich jak *Problemy według ważności* lub *Hosty problemowe* itd., uwzględniane są tylko problemy przyczynowe. Jednak klasyfikacja problemów nie wpływa na usługi.

Problem objawowy może być powiązany tylko z jednym problemem przyczynowym. Problemy objawowe nie są rozwiązywane automatycznie, jeśli problem przyczynowy zostanie rozwiązany lub zamknięty.

Konfiguracja

Aby przeklasyfikować problem jako problem objawu, najpierw zaznacz go na liście problemów. Można zaznaczyć jeden lub kilka problemów.

Następnie przejdź do problemu przyczyny i w jego menu kontekstowym kliknij opcję *Oznacz zaznaczone jako objawy*.



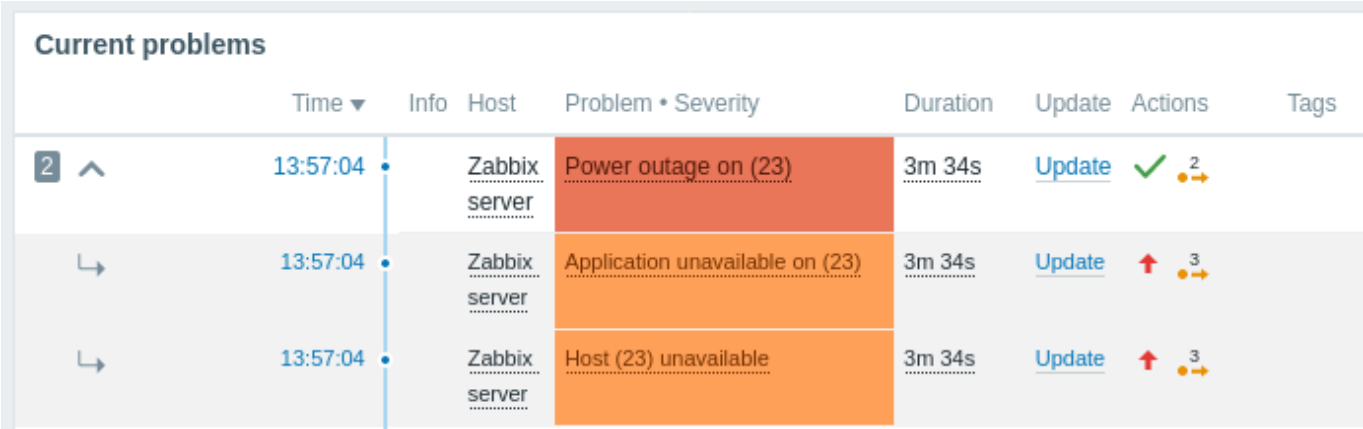
Po tym zaznaczone problemy zostaną zaktualizowane przez serwer do problemów objawów problemu przyczyny.

Podczas aktualizowania statusu problemu jest on wyświetlany na jeden z dwóch sposobów:

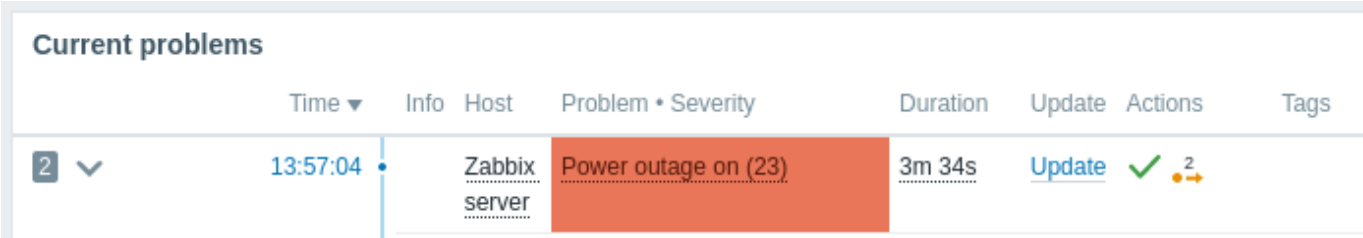
- W kolumnie Status wyświetlany jest migający status "UPDATING";
- W kolumnie Info wyświetlana jest migająca ikona lub (jest to stosowane, jeśli w filtrze wybrano tylko *Problems*, a więc kolumna Status nie jest wyświetlana).

Wyświetlanie

Problemy objawowe są wyświetlane poniżej problemu przyczynowego i odpowiednio oznaczone w *Monitoring* > *Problems* (oraz w widżecie pulpitu *Problems*) - za pomocą ikony, mniejszej czcionki i innego tła.



W widoku zwiniętym widoczny jest tylko problem przyczynowy; o istnieniu problemów objawowych informuje liczba na początku wiersza oraz ikona rozwijania widoku.



Możliwe jest także dodatkowe wyświetlanie problemów objawowych zwykłą czcionką i w osobnym wierszu. W tym celu należy wybrać *Show symptoms* w ustawieniach filtra lub konfiguracji widżetu.

Przywracanie do problemu przyczynowego

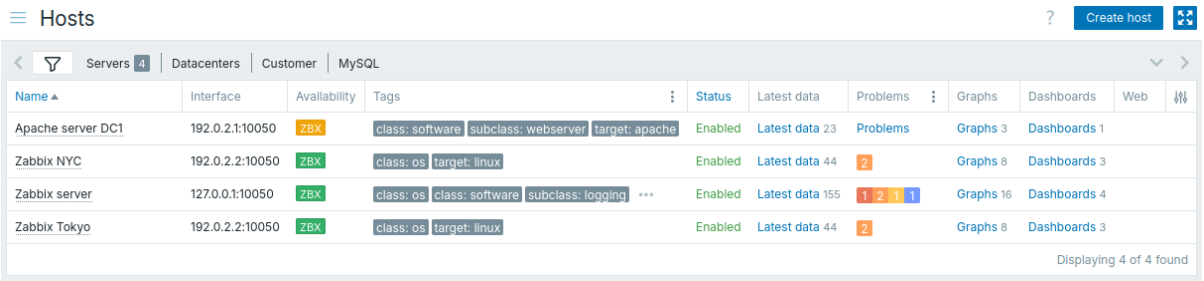
Problem objawowy można przywrócić do problemu przyczynowego. Aby to zrobić, należy:

- kliknąć opcję *Oznacz jako przyczynę* w menu kontekstowym problemu objawowego;
- zaznaczyć opcję *Konwertuj na przyczynę* na ekranie aktualizacji problemu i kliknąć *Aktualizuj* (ta opcja będzie działać również wtedy, gdy zaznaczono kilka problemów).

Hosty

Omówienie

Sekcja *Monitoring* > *Hosts* wyświetla pełną listę monitorowanych hostów z szczegółowymi informacjami o interfejsie hosta, dostępności, tagach, bieżących problemach, statusie (włączony/wyłączony) oraz linkami umożliwiającymi łatwe przechodzenie do najnowszych danych hosta, historii problemów, wykresów, pulpitów i scenariuszy web.



Szczegóły hostów są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia pokazywanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie układu domyślnego. Możesz także zmieniać szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślny rozmiar. Wszystkie dostosowania są stosowane natychmiast i są zawsze zapamiętywane dla bieżącego użytkownika w aktywnej karcie filtra (domyślnej lub zapisanej).

Dostępne są następujące kolumny:

Column	Description
Name	Widoczna nazwa hosta. Kliknięcie nazwy otwiera menu hosta. Pomarańczowa ikona klucza po nazwie oznacza, że host znajduje się w konserwacji. Kliknij nagłówek kolumny, aby sortować hosty według nazwy rosnąco lub malejąco.

Column	Description
Interface	Wyświetlany jest główny interfejs hosta.
Availability	Wyświetlana jest dostępność hosta dla skonfigurowanych interfejsów. Ikony dostępności reprezentują bieżący status interfejsu hosta na serwerze Zabbix. Dlatego jeśli wyłączysz hosta we frontend, jego dostępność zostanie zaktualizowana po tym, jak serwer Zabbix zsynchronizuje zmiany konfiguracji. Podobnie, jeśli włączysz hosta, jego dostępność zostanie zaktualizowana po tym, jak serwer Zabbix zsynchronizuje zmiany konfiguracji i odpyta hosta. Ikony dostępności reprezentują tylko te typy interfejsów (Agent, SNMP, IPMI, JMX), które są skonfigurowane. Po najechaniu kursorem na ikonę wyświetla się okno podręczne z listą wszystkich interfejsów tego samego typu wraz ze szczegółami, statusem i błędami. Dla interfejsu Agent okno podręczne wyświetla interfejsy (pasywne) oraz aktywne sprawdzenia. Jeśli host ma tylko aktywne sprawdzenia, ikona interfejsu Agent jest wyświetlana nawet wtedy, gdy host nie ma skonfigurowanego interfejsu Agent. Kolumna jest pusta dla hostów bez interfejsów. Status pojedynczego interfejsu hosta jest określany na podstawie połączenia między włączoną pozycją korzystającą z tego interfejsu a hostem. Status może być następujący: **Available** - połączenie z hostem powiodło się; **Not available** - połączenie z hostem nie powiodło się (przekroczenie limitu czasu, problemy z zaporą sieciową itp.); **Unknown** - nie podjęto próby połączenia z hostem lub wynik jest nieznany. Więcej informacji o tym, jak serwer Zabbix określa status interfejsu, można znaleźć w sekcjach Unknown interface status oraz Unreachable/unavailable host interface settings. Status wszystkich interfejsów hosta jednego typu (Agent, SNMP, IPMI, JMX) jest określany na podstawie tych interfejsów, które są używane przez co najmniej jedną włączoną pozycję. Status jest oznaczany kolorem ikony: **Green** - wszystkie interfejsy są dostępne; **Yellow** - co najmniej jeden interfejs jest niedostępny, a co najmniej jeden jest dostępny lub nieznany; **Red** - wszystkie interfejsy są niedostępne; **Gray** - co najmniej jeden interfejs jest nieznany, ale żaden nie jest niedostępny. **Dostępność aktywnych sprawdzeń.** Jeśli na hoście włączono co najmniej jedno aktywne sprawdzenie, aktywne sprawdzenia również wpływają na łączną dostępność interfejsu Agent, jak opisano powyżej. Aby określić dostępność aktywnych sprawdzeń, komunikaty heartbeat są wysyłane w wątku aktywnych sprawdzeń agenta. Częstotliwość komunikatów heartbeat jest □□□□□□□owana przez parametr `HeartbeatFrequency` w konfiguracji Zabbix agent lub agent 2 (domyślnie 60 sekund, zakres 0-3600). Aktywne sprawdzenia są uznawane za niedostępne, gdy heartbeat aktywnego sprawdzenia jest starszy niż 2 x HeartbeatFrequency sekund. **Uwaga:** Agenty Zabbix starsze niż wersja 6.2.x nie wysyłają heartbeat aktywnych sprawdzeń, więc dostępność ich hostów pozostaje nieznana.
Tags	Wyświetlane są tagi hosta. Po najechaniu kursorem lub kliknięciu ikony trzech kropek po wyświetlonych tagach otwiera się okno podręczne ze wszystkimi tagami. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia ustawienie liczby wyświetlanych tagów (1–3), wyboru sposobu wyświetlania nazw tagów (pełne, skrócone lub brak), ustawienia priorytetu wyświetlania tagów lub duplikowanie kolumny (zduplikowane kolumny można zmieniać nazwę lub usuwać).
Tag value	Wyświetlane są wartości tagów hosta. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia zmianę nazwy kolumny, wybór wyświetlanych wartości tagów lub duplikowanie kolumny (zduplikowane kolumny można zmieniać nazwę lub usuwać).
Status	Status hosta - *Enabled* lub *Disabled*. Kliknij nagłówek kolumny, aby sortować hosty według statusu rosnąco lub malejąco.
Latest data	Kliknięcie linku otworzy sekcję *Monitoring* > *Latest data* ze wszystkimi najnowszymi danymi zebranymi z hosta. Liczba pozycji z najnowszymi danymi jest wyświetlana na szaro.

Column	Description
Problems	Wyświetlana jest liczba otwartych problemów dla hosta, posortowana według ważności. Kolor każdej ikony problemu pokazuje ważność, a liczba pokazuje liczbę problemów dla danej ważności. Kliknięcie ikony otwiera *Monitoring* > *Problems* dla tego hosta; jeśli host nie ma problemów, zamiast tego wyświetlany jest link *Problems*. Kliknięcie przycisku *Opcje* w nagłówku kolumny daje dostęp do opcji *Show suppressed problems*, która obejmuje problemy, które w przeciwnym razie byłyby tłumione (ukryte) z powodu konserwacji hosta lub pojedynczego tłumienia problemu.
Graphs	Kliknięcie linku wyświetli wykresy skonfigurowane dla hosta. Liczba wykresów jest wyświetlana na szaro. Jeśli host nie ma wykresów, link nie jest wyświetlany.
Dashboards	Kliknięcie linku wyświetli pulpity skonfigurowane dla hosta. Liczba pulpitów jest wyświetlana na szaro. Jeśli host nie ma pulpitów, link nie jest wyświetlany.
Web	Kliknięcie linku wyświetli scenariusze web skonfigurowane dla hosta. Liczba scenariuszy web jest wyświetlana na szaro. Jeśli host nie ma scenariuszy web, link nie jest wyświetlany.
Custom text	Wyświetlany jest tekst niestandardowy, wartości makr lub ich kombinacja. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia zmianę nazwy kolumny, wprowadzenie tekstu niestandardowego lub makr albo duplikowanie kolumny (zduplikowane kolumny można zmieniać nazwę lub usuwać). Obsługiwane makra: `{HOST.*}` (z wyjątkiem `{HOST.METADATA}`), `{INVENTORY.*}` oraz makra użytkownika.

Przyciski

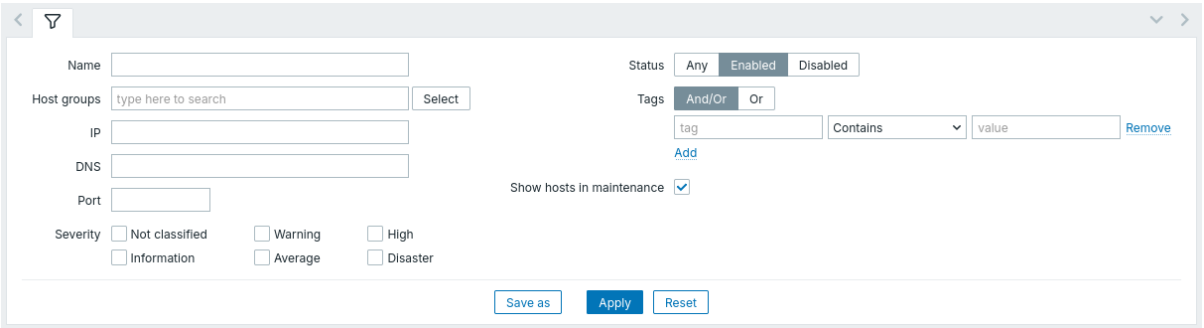
Utwórz host umożliwia utworzenie nowego hosta. Ten przycisk jest dostępny tylko dla użytkowników z rolą Administratora i Super administratora.

Przyciski trybu widoku wspólne dla wszystkich sekcji opisano na stronie Monitorowanie.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko hosty, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są wyszukiwane z nierozwiniętymi makrami.

Filtr znajduje się nad tabelą. Można filtrować hosty według nazwy, grupy hostów, IP lub DNS, portu interfejsu, tagów, poziomu problemu, statusu (włączony/wyłączony/dowolny); można też wybrać, czy mają być wyświetlane hosty, które są obecnie w konserwacji.



Parameter	Description
Name	Filtruj według widocznej nazwy hosta.
Host groups	Filtruj według jednej lub kilku grup hostów. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów.
IP	Filtruj według adresu IP.
DNS	Filtruj według nazwy DNS.
Port	Filtruj według numeru portu.
Severity	Filtruj według poziomu problemu. Domyślnie wyświetlane są problemy wszystkich poziomów. Problemy są wyświetlane, jeśli nie są wyciszone.
Status	Filtruj według statusu hosta.

Parameter	Description
Tags	Filtruj według nazwy i wartości tagu hosta. Hosty można filtrować zarówno według tagów na poziomie hosta, jak i tagów ze wszystkich powiązanych szablonów, w tym zagnieżdżonych szablonów. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek
Show hosts in maintenance	Zaznacz pole wyboru, aby wyświetlić hosty, które są w konserwacji (wyświetlane domyślnie).

Zapisywanie filtra

Ustawienia ulubionego filtra można zapisać jako karty, a następnie szybko uzyskiwać do nich dostęp, klikając odpowiednią kartę nad filtrem.

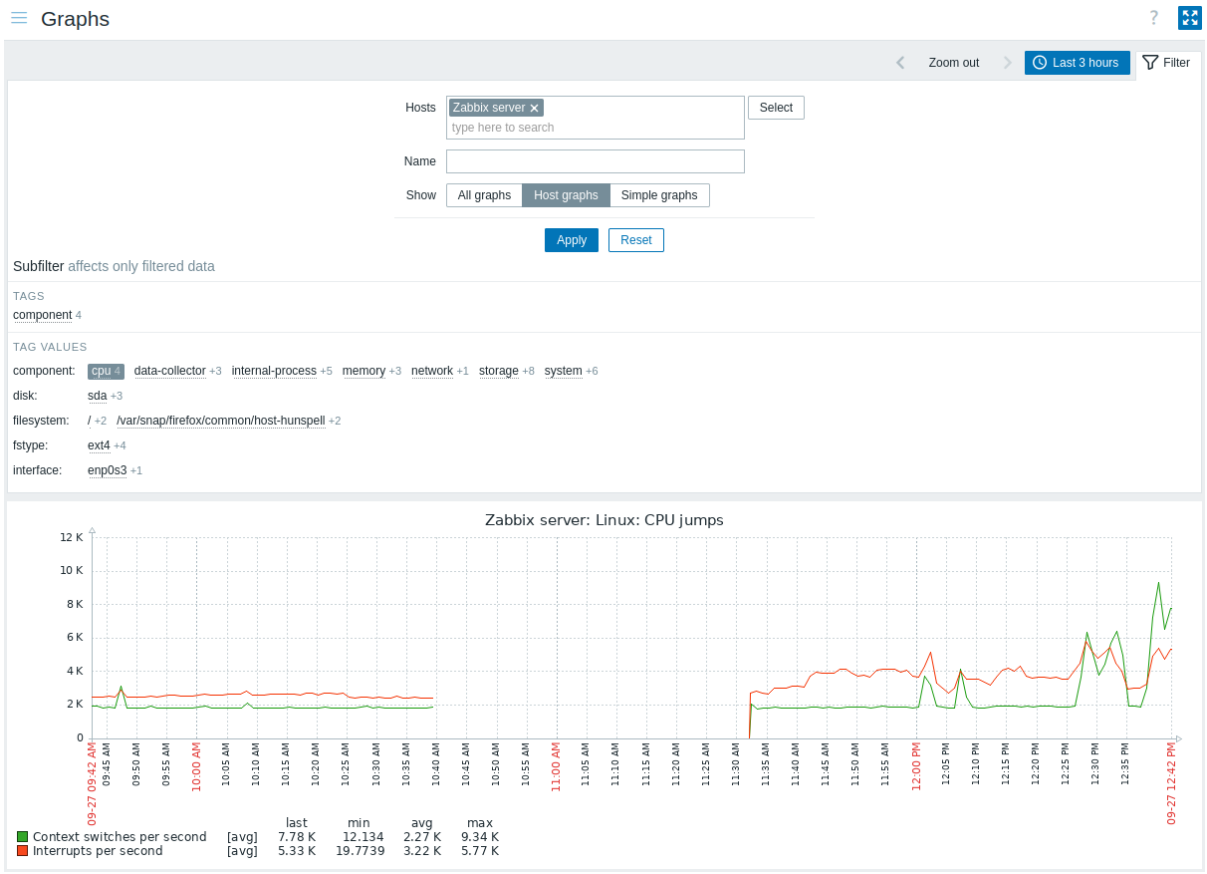
Więcej informacji znajdziesz w sekcji zapisywanie filtrów.

Wykresy

Przegląd

Do wykresów hosta można uzyskać dostęp z poziomu *Monitoring* > *Hosts*, klikając Wykresy dla odpowiedniego hosta.

Można wyświetlić dowolny niestandardowy wykres skonfigurowany dla hosta, a także dowolny prosty wykres.



Wykresy są sortowane według:

- nazwy wykresu (niestandardowe wykresy)
- nazwy pozycji (proste wykresy)

Dostępne są również wykresy dla wyłączonych hostów.

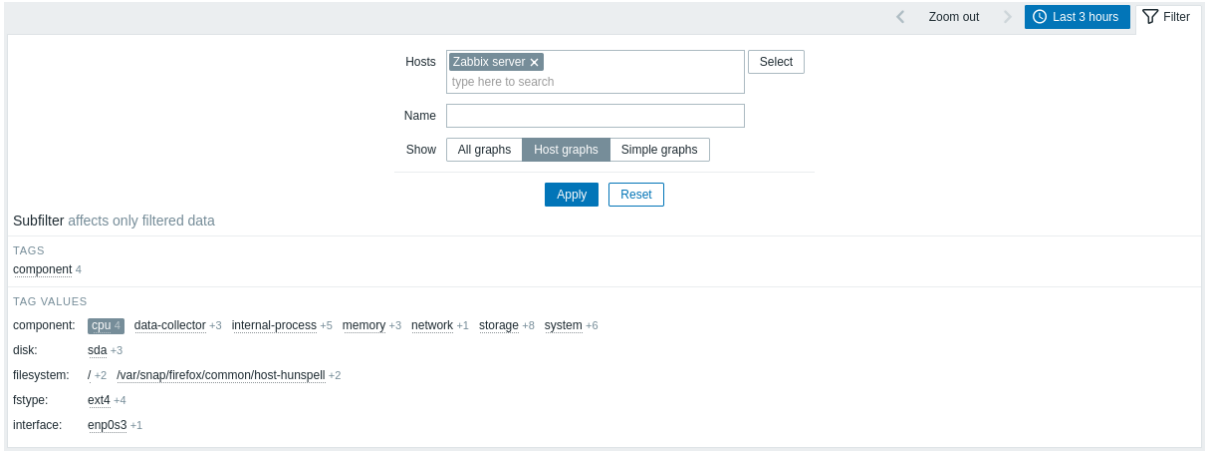
Selektor przedziału czasu

Zwróć uwagę na selektor przedziału czasu nad wykresem. Umożliwia on wybór często wymaganych przedziałów jednym kliknięciem myszy.

Zobacz także: *Selektor przedziału czasu*

Korzystanie z filtra

Aby wyświetlić określony wykres, wybierz go w filtrze. Filtr umożliwia określenie hosta, nazwy wykresu oraz opcji *Pokaż* (wszystkie/wykresy hosta/wykresy proste).



Jeśli w filtrze nie wybrano żadnego hosta, wykresy nie są wyświetlane.

Korzystanie z podfiltra

Podfiltr umożliwia dalszą modyfikację filtrowania zastosowanego przez filtr główny.

Zawiera klikalne odnośniki zapewniające szybki dostęp do powiązanych wykresów. Wykresy są powiązane wspólną encją - nazwą lub wartością tagu. Po kliknięciu nazwy lub wartości tagu zostaje ona wyróżniona szarym tłem, a wykresy są natychmiast filtrowane (nie trzeba klikać przycisku *Zastosuj* w filtrze głównym). Kliknięcie innej nazwy lub wartości tagu dodaje ją do filtrowanych wyników. Ponowne kliknięcie nazwy lub wartości tagu usuwa filtrowanie.

Podfiltry są generowane na podstawie przefiltrowanych danych, które są ograniczone do 1000 rekordów. Aby wyświetlić więcej rekordów w podfiltrze, należy zwiększyć wartość parametru *Limit wyników wyszukiwania i filtrowania* (w sekcji *Administracja* > *Ogólne* > *GUI*).

W przeciwieństwie do filtra głównego podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze wyświetlać aktualne informacje o dostępnych opcjach filtrowania i ich licznikach.

Liczba encji wyświetlanych w poziomie jest ograniczona do 100. Jeśli jest ich więcej, na końcu wyświetlana jest ikona trzech kropek; nie można jej kliknąć. Listy pionowe (takie jak tagi wraz z ich wartościami) są ograniczone do 20 wpisów. Jeśli jest ich więcej, wyświetlana jest ikona trzech kropek; nie można jej kliknąć.

Liczba obok każdej klikalnej encji wskazuje liczbę wykresów, które zawiera ona w wynikach filtra głównego.

Po wybraniu jednej encji liczby przy innych dostępnych encjach są wyświetlane ze znakiem plus, wskazującym, ile wykresów można dodać do bieżącego wyboru.

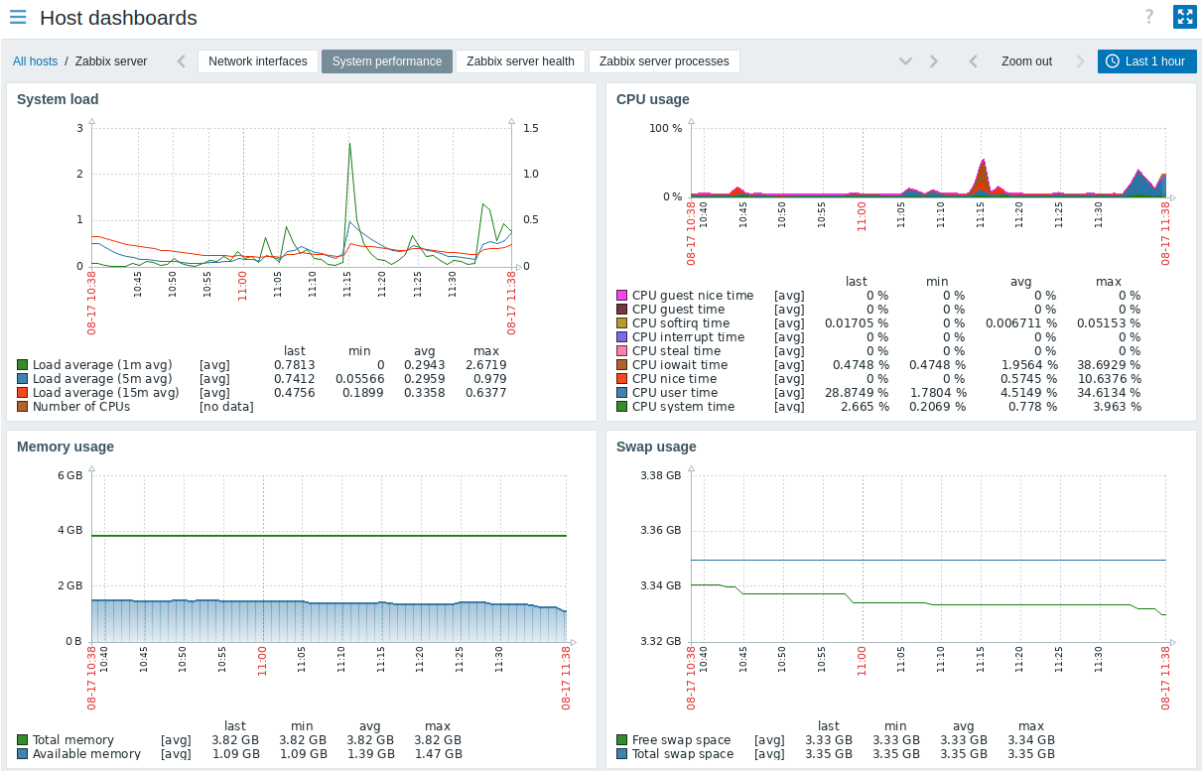
Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, są opisane na stronie Monitoring.

Pulpity hostów

Przegląd

Pulpity hostów wyglądają podobnie do pulpitów globalnych; jednak pulpity hostów nie mają właściciela i wyświetlają dane tylko dla wybranego hosta.



Podczas wyświetlania pulpitów hostów można przełączać się między skonfigurowanymi pulpitami, klikając:

- karty pulpitów;
- przyciski strzałek pod nagłówkiem;
- przycisk strzałki pod nagłówkiem, który wyświetli pełną listę dostępnych pulpitów hostów.

Aby przejść do sekcji *Monitoring* > *Hosts*, kliknij łącze nawigacyjne *All hosts* pod nagłówkiem w lewym górnym rogu.

Konfiguracja

Pulpity hostów są konfigurowane na poziomie szablonu. Po podłączeniu szablonu do hosta dla tego hosta generowane są pulpity hosta. Należy pamiętać, że pulpitów hosta *nie można* konfigurować w sekcji *Pulpity*, która jest zarezerwowana dla pulpitów globalnych.

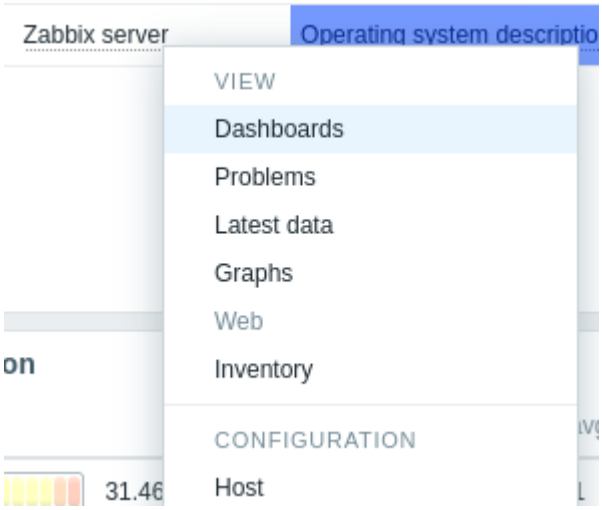
Widżety pulpitów hosta również można konfigurować wyłącznie na poziomie szablonu, z wyjątkiem zmiany interwału odświeżania. Ponadto widżety pulpitów hosta można kopiować wyłącznie do innych pulpitów hosta w obrębie tego samego szablonu. Należy pamiętać, że widżetów z pulpitów globalnych *nie można* kopiować do pulpitów hosta.

Note:
Pulpity hosta przed Zabbix 5.2 były ekranami hosta. Podczas importowania starszego szablonu zawierającego ekrany import ekranów zostanie zignorowany.

Dostęp

Do pulpitów hosta można uzyskać dostęp:

- po wyszukaniu nazwy hosta w wyszukiwaniu globalnym (kliknij link *Pulpity* dostępny w wynikach wyszukiwania);
- po kliknięciu nazwy hosta w *Inventory* > *Hosts* (kliknij link *Pulpity* dostępny w podglądzie hosta);
- z menu hosta, klikając *Pulpity*.

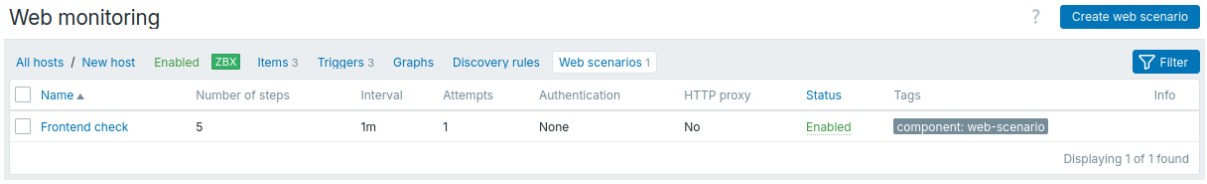


Należy pamiętać, że do pulpitów hosta *nie można* uzyskać bezpośredniego dostępu w sekcji *Pulpity*, która jest zarezerwowana dla globalnych pulpitów.

Scenariusze webowe

Przegląd

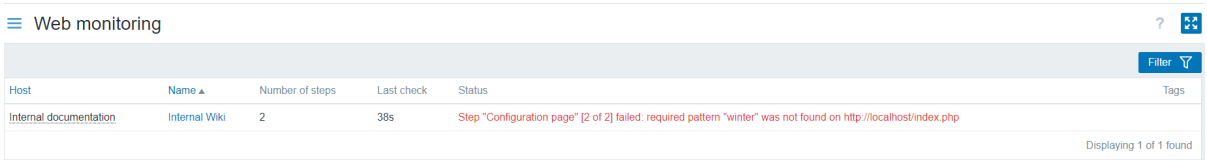
Informacje o scenariuszu web hosta można wyświetlić w *Monitoring* > *Hosts*, klikając Web dla odpowiedniego hosta.



Kliknięcie nazwy hosta wyświetla menu hosta. Dane wyłączonych hostów są również dostępne. Nazwa wyłączonego hosta jest wyświetlana na czerwono.

Status scenariusza web może być następujący:

- OK (zielony) - wszystkie kroki zakończyły się powodzeniem, a Zabbix zebrał dane dla scenariusza.
- Błąd (czerwony) - jeden lub więcej kroków zakończyło się niepowodzeniem (zobacz komunikat o błędzie, aby uzyskać wskazówki, które parametry mogą wymagać poprawy).
- Nieznany (szary) - status żadnego kroku nie jest jeszcze dostępny lub nie zebrano jeszcze żadnych danych.



Maksymalna liczba scenariuszy wyświetlanych na stronie zależy od ustawienia profilu użytkownika *Rows per page* setting.

Domyślnie wyświetlane są tylko wartości mieszczące się w ostatnich 24 godzinach. To ograniczenie wprowadzono w celu skrócenia czasu początkowego ładowania dużych stron z najnowszymi danymi. Możesz wydłużyć ten okres, zmieniając wartość parametru *Max history display period* w sekcji menu *Administration > General > GUI*.

Nazwa scenariusza jest linkiem do bardziej szczegółowych statystyk na jego temat:



Używanie filtra

Strona wyświetla listę wszystkich scenariuszy web wybranego hosta. Aby wyświetlić scenariusze web dla innego hosta lub grupy hostów bez powracania do strony *Monitoring > Hosts*, wybierz ten host lub grupę w filtrze. Możesz także filtrować scenariusze na podstawie tagów.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Najnowsze dane

Omówienie

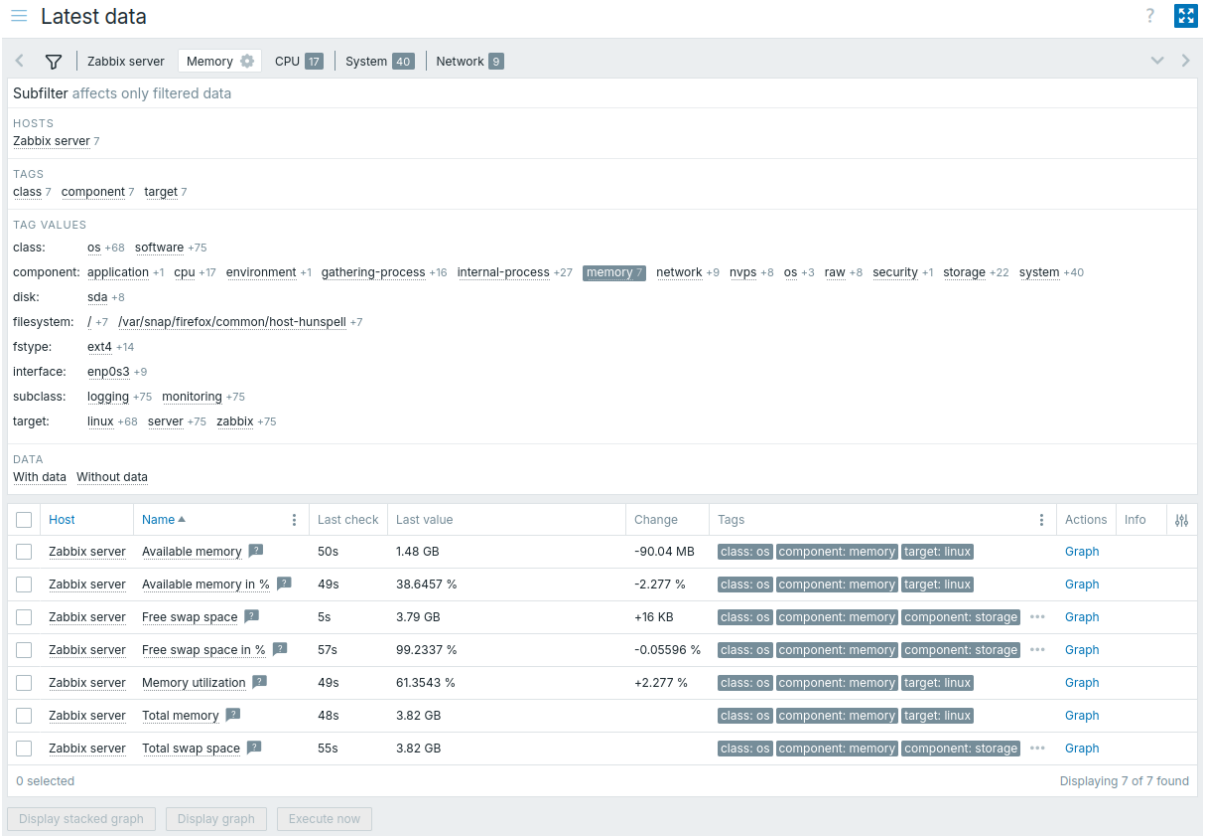
Sekcja *Monitoring > Latest data* wyświetla najnowsze wartości zebrane przez pozycje.

Ta sekcja zawiera następujące elementy:

- Filtr

- Subfiltr
- Lista pozycji

Note:
Subfiltr i lista pozycji są wyświetlane tylko wtedy, gdy filtr jest ustawiony i istnieją wyniki do wyświetlenia.



Szczegóły najnowszych wartości są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia pokazywanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie układu domyślnego. Możesz także zmieniać szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca ustawienie domyślne. Wszystkie dostosowania są stosowane natychmiast i są zawsze zapamiętywane dla bieżącego użytkownika w aktywnej karcie filtra (domyślnej lub zapisanej).

Dostępne są następujące kolumny:

Column	Description
Host	Nazwa hosta, do którego należy pozycja. Kliknięcie nazwy otwiera menu kontekstowe hosta. Jeśli host jest w trybie konserwacji, po nazwie hosta wyświetlana jest pomarańczowa ikona klucza . Jeśli host jest wyłączony, jego nazwa jest wyświetlana na czerwono. Należy pamiętać, że dane wyłączonych hostów (w tym wykresy i listy wartości pozycji) są dostępne w sekcji *Latest data*.
Name	Nazwa pozycji. Kliknięcie nazwy otwiera menu pozycji. Obok nazwy pozycji dla wszystkich pozycji, które mają opis, wyświetlana jest ikona znaku zapytania . Najedź kursorem na ikonę, aby wyświetlić podpowiedź z opisem pozycji. Kliknięcie przycisku *Options* w nagłówku kolumny daje dostęp do opcji *Show item key*, która wyświetla klucz pozycji pod nazwą każdej pozycji.
Interval	Interwał aktualizacji pozycji.
History	Okres przechowywania historii pozycji.
Trends	Okres przechowywania trendów pozycji.
Type	Typ pozycji.
Last check	Czas od ostatniego sprawdzenia pozycji.

Column	Description
Last value	Najnowsza wartość pozycji. Wartości są wyświetlane z zastosowaną konwersją jednostek i mapowaniem wartości. Najedź kursorem na wartość, aby wyświetlić surowe dane (przycięte do 8192 znaków). Wartości binarne są wyświetlane jako symbol zastępczy (*binary value*) zamiast rzeczywistej wartości. Domyślnie wyświetlane są tylko wartości odebrane w ciągu ostatnich 24 godzin. To ograniczenie poprawia czas początkowego ładowania dużych stron z najnowszymi danymi; aby je rozszerzyć, zmień wartość parametru *Max history display period* w *Administration* > *General* > *GUI*.
Change	Różnica między poprzednią wartością a najnowszą wartością. W przypadku pozycji z częstotliwością aktualizacji wynoszącą 1 dzień lub więcej, wartość zmiany nigdy nie będzie wyświetlana (przy ustawieniach domyślnych). W takim przypadku najnowsza wartość nie będzie w ogóle wyświetlana, jeśli została odebrana ponad 24 godziny temu.
Tags	Tagi powiązane z pozycją. Kliknięcie tagu włącza go w subfiltrze, filtrując w ten sposób listę pozycji według tego tagu (i dowolnych innych tagów wybranych w subfiltrze). Po kliknięciu takie tagi nie są już klikalne; aby dostosować filtrowanie według tagów, należy użyć subfiltra. Kliknięcie przycisku *Options* w nagłówku kolumny pozwala ustawić liczbę wyświetlanych tagów (1-3), wybrać sposób wyświetlania nazw tagów (pełny, skrócony lub brak), ustawić priorytet wyświetlania tagów albo zduplikować kolumnę (zduplikowane kolumny można zmieniać nazwę lub usuwać).
Tag value	Wyświetlane są wartości tagów pozycji. Kliknięcie przycisku *Options* w nagłówku kolumny pozwala zmienić nazwę kolumny, wybrać, które wartości tagów mają być wyświetlane, albo zduplikować kolumnę (zduplikowane kolumny można zmieniać nazwę lub usuwać).
Actions	Link do prostego wykresu/historii wartości pozycji.
Info	Dodatkowe informacje o pozycji. Jeśli pozycja zawiera błędy (na przykład stała się nieobsługiwana), wyświetlana jest ikona informacji. Najedź kursorem na ikonę, aby zobaczyć szczegóły.
Custom text	Wyświetlany jest tekst niestandardowy, wartości makr lub ich kombinacja. Kliknięcie przycisku *Options* w nagłówku kolumny pozwala zmienić nazwę kolumny, wprowadzić tekst niestandardowy lub makra albo zduplikować kolumnę (zduplikowane kolumny można zmieniać nazwę lub usuwać). Obsługiwane makra: `{HOST.*}` (z wyjątkiem `{HOST.METADATA}`), `{INVENTORY.*}`, `{ITEM.*}` (z wyjątkiem `{ITEM.STATE.ERROR}`) oraz makra użytkownika.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, opisano na stronie Monitoring.

Akcje masowe

Przyciski poniżej listy oferują akcje masowe dla jednego lub kilku zaznaczonych elementów:

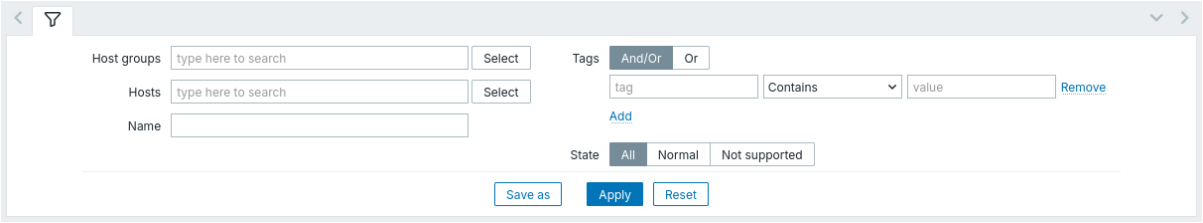
- *Wyświetl wykres skumulowany* - wyświetla skumulowany wykres ad hoc.
- *Wyświetl wykres* - wyświetla prosty wykres ad hoc.
- *Wykonaj teraz* - natychmiast wykonuje sprawdzenie nowych wartości pozycji. Obsługiwane tylko dla sprawdzeń **pasywnych** (zobacz więcej szczegółów). Ta opcja jest dostępna tylko dla hostów z dostępem do odczytu i zapisu. Dostęp do tej opcji dla hostów z uprawnieniami tylko do odczytu zależy od opcji roli użytkownika o nazwie *Wywołaj "Wykonaj teraz" na hostach tylko do odczytu*.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich pozycjach, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko interesujące Cię pozycje. Aby zwiększyć wydajność wyszukiwania, dane są wyszukiwane bez rozwiniętych makr.

Ikona filtra  znajduje się nad listą pozycji i podfiltrem. Kliknij ją, aby rozwinąć filtr.



Filtr umożliwia zawężenie listy pozycji według grupy hostów, hosta, nazwy pozycji, znaczników i stanu. Określenie nadrzędnej grupy hostów w filtrze powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. Szczegółowe informacje na temat filtrowania według znaczników znajdziesz w sekcji *Monitoring* > *Problems*.

Opcja filtra *Show details* umożliwia wyświetlenie dodatkowych informacji o pozycjach. Zaznacz ją, aby wyświetlić takie szczegóły jak interwał odświeżania pozycji, ustawienia historii i trendów, typ pozycji oraz błędy pozycji (fine/unsupported).

Zapisywanie filtra

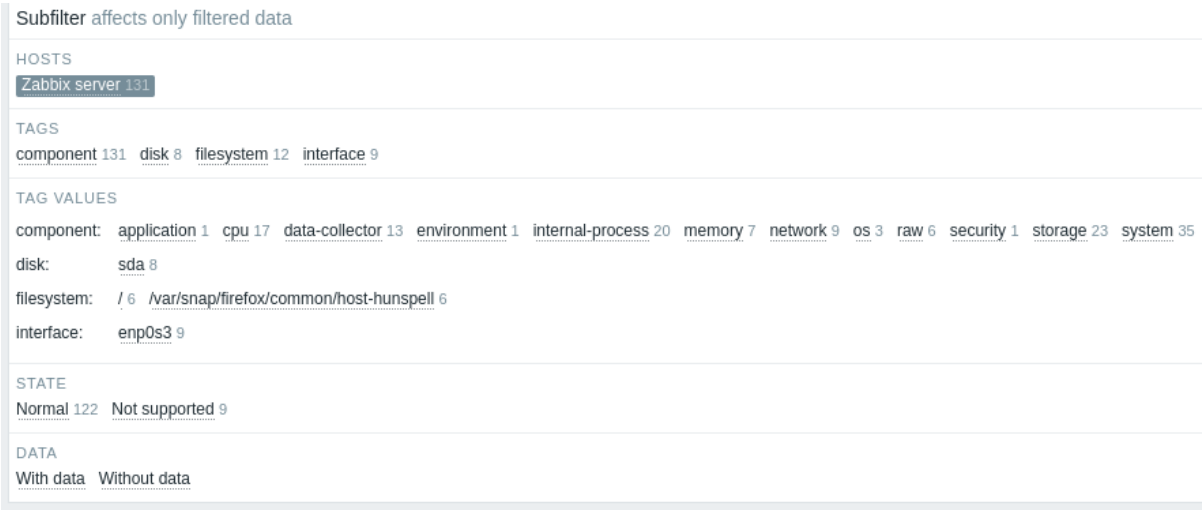
Ustawienia ulubionego filtra można zapisać jako karty, a następnie szybko uzyskać do nich dostęp, klikając odpowiednią kartę nad filtrem.

Więcej informacji znajdziesz w sekcji zapisywania filtrów.

Korzystanie z podfiltra

Podfiltr umożliwia dalsze modyfikowanie filtrowania z filtra głównego.

Zawiera klikalne odnośniki zapewniające szybki dostęp do powiązanych elementów. Elementy są powiązane za pomocą wspólnej encji - hosta, nazwy lub wartości tagu, stanu pozycji albo stanu danych. Po kliknięciu encji zostaje ona wyróżniona szarym tłem, a pozycje są natychmiast filtrowane (nie trzeba klikać przycisku *Zastosuj* w filtrze głównym). Kliknięcie innej encji dodaje ją do przefiltrowanych wyników. Ponowne kliknięcie encji usuwa filtrowanie.



Podfiltry są generowane na podstawie przefiltrowanych danych, które są ograniczone do 1000 rekordów. Jeśli istnieje 20 hostów, z których każdy ma 100 pozycji (czyli łącznie 2000 rekordów), w podfiltrze będzie widoczna tylko połowa hostów. Aby wyświetlić więcej rekordów w podfiltrze, należy zwiększyć wartość parametru *Limit for search and filter results* (w sekcji *Administracja* > *Ogólne* > *GUI*).

W przeciwieństwie do filtra głównego podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zawierał aktualne informacje o dostępnych opcjach filtrowania i odpowiadających im licznikach.

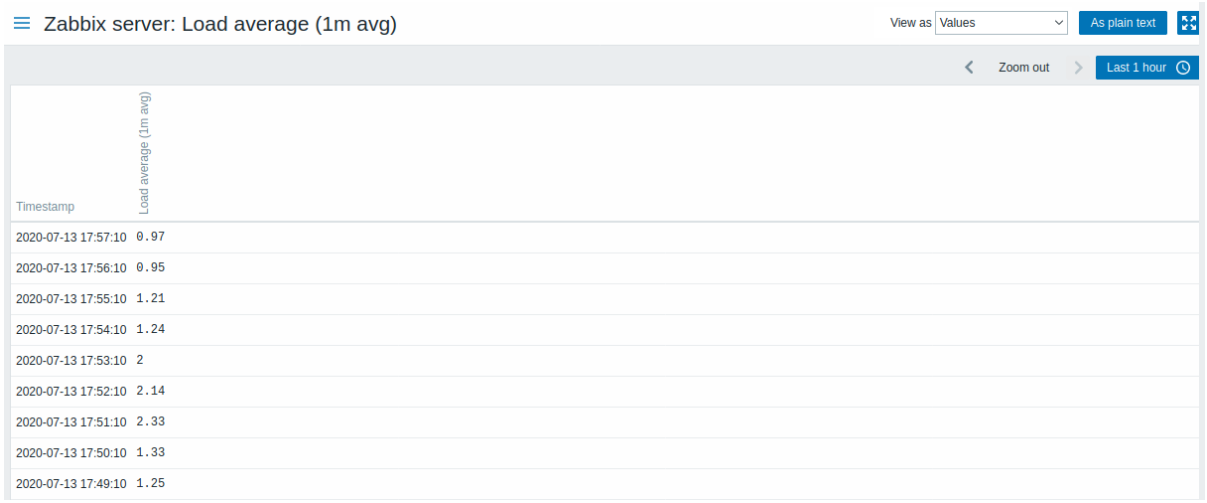
Dla każdej grupy encji (hostów, tagów, wartości tagów itd.) wyświetlanych jest maksymalnie 10 wierszy encji. Jeśli istnieje więcej encji, listę można rozwinąć, aby wyświetlić maksymalnie 1000 wpisów (wartość SUBFILTER_VALUES_PER_GROUP w definicjach frontendu), klikając ikonę z trzema kropkami ••• na końcu listy. W przypadku *Wartości tagów* listę można rozwinąć, aby wyświetlić maksymalnie 200 nazw tagów wraz z odpowiadającymi im wartościami. Należy pamiętać, że po pełnym rozwinięciu listy nie można jej zwinąć.

Liczba obok każdej klikalnej encji wskazuje liczbę pogrupowanych w niej pozycji (na podstawie wyników filtra głównego). Po kliknięciu encji liczby przy innych dostępnych encjach są wyświetlane ze znakiem plus, wskazującym, ile pozycji można dodać do bieżącego wyboru. Encje bez pozycji nie są wyświetlane, chyba że wcześniej wybrano je w podfiltrze.

Wykresy i historia

Kolumna *Graph/History* na liście pozycji oferuje następujące linki:

- **History** - dla wszystkich tekstowych pozycji, prowadzący do list (*Values/500 latest values*) wyświetlających historię poprzednich wartości pozycji.
- **Graph** - dla wszystkich numerycznych pozycji, prowadzący do prostego wykresu. Należy pamiętać, że gdy wykres jest wyświetlany, lista rozwijana w prawym górnym rogu oferuje również możliwość przełączenia na *Values/500 latest values*.



Wartości wyświetlane na tej liście są surowe, to znaczy nie jest stosowane żadne przetwarzanie końcowe.

Note:
Całkowita liczba wyświetlanych wartości jest określana przez wartość parametru *Limit for search and filter results*, ustawionego w *Administration* > *General* > *GUI*.

Mapy

Przegląd

W sekcji *Monitoring* > *Maps* można konfigurować, zarządzać i wyświetlać mapy sieci.

Po otwarciu tej sekcji zobaczysz albo ostatnią mapę, do której uzyskałeś dostęp, albo listę wszystkich map, do których masz dostęp.

Wszystkie mapy mogą być publiczne lub prywatne. Mapy publiczne są dostępne dla wszystkich użytkowników, natomiast mapy prywatne są dostępne tylko dla ich właściciela oraz użytkowników, z którymi mapa została udostępniona.

Lista map



Wyświetlane dane:

Column	Description
Name	Nazwa mapy. Kliknij nazwę, aby wyświetlić mapę.
Width	Wyświetlana jest szerokość mapy.
Height	Wyświetlana jest wysokość mapy.
Actions	Dostępne są dwie akcje: **Properties** - ustaw ogólne właściwości mapy **Edit** - uzyskaj dostęp do siatki, aby dodawać elementy mapy

Aby skonfigurować nową mapę, kliknij przycisk *Create map* w prawym górnym rogu. Aby zaimportować mapę z pliku YAML, XML lub JSON, kliknij przycisk *Import* w prawym górnym rogu. Użytkownik, który importuje mapę, zostanie ustawiony jako jej właściciel.

Dwa przyciski pod listą oferują opcje masowej edycji:

- *Export* - eksportuj mapy do pliku YAML, XML lub JSON
- *Delete* - usuń mapy

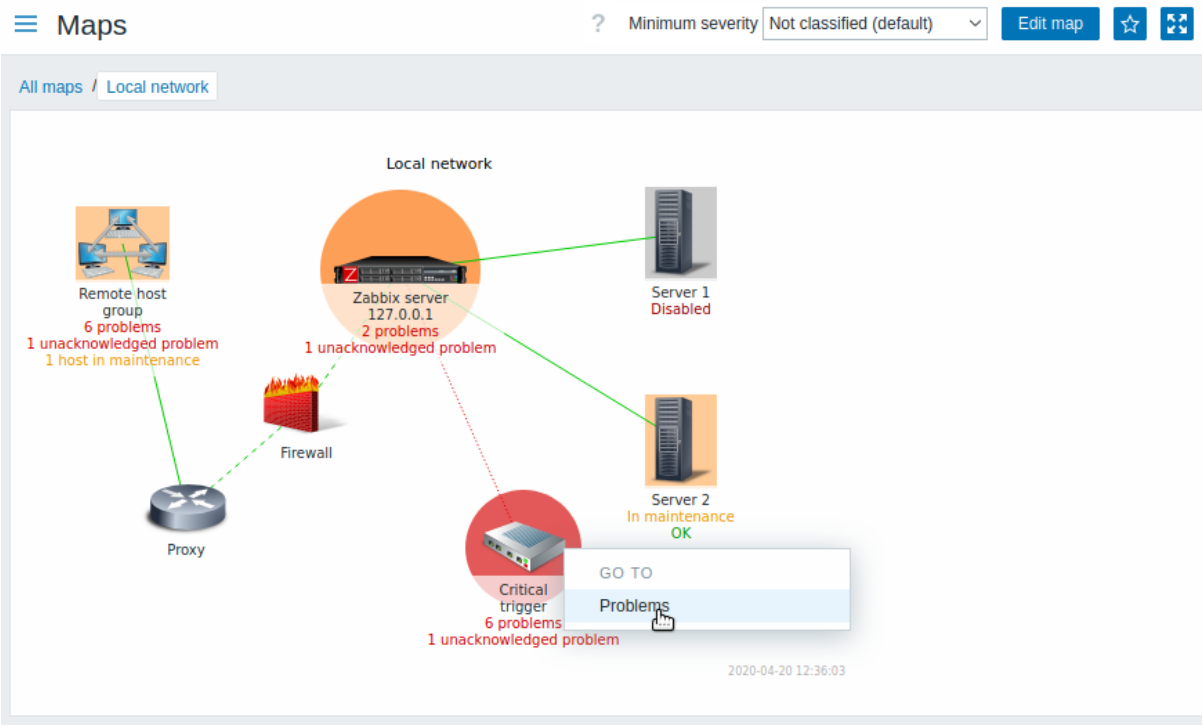
Aby skorzystać z tych opcji, zaznacz pola wyboru przed odpowiednimi mapami, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko mapy, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Wyświetlanie map

Aby wyświetlić mapę, kliknij jej nazwę na liście wszystkich map.



Możesz użyć listy rozwijanej na pasku tytułu mapy, aby wybrać najniższy poziom ważności wyzwalaczy problemów do wyświetlenia. Ważność oznaczona jako *default* to poziom ustawiony w konfiguracji mapy. Jeśli mapa zawiera podmapę, przejście do podmapy zachowa ważność mapy nadrzędnej (z wyjątkiem sytuacji, gdy jest to *Not classified*; w takim przypadku nie zostanie ona przekazana do podmapy).

Wyróżnianie ikon

Jeśli element mapy znajduje się w stanie problemu, jest wyróżniany za pomocą okręgu. Kolor wypełnienia okręgu odpowiada kolorowi ważności problemu. Tylko problemy o wybranym poziomie ważności lub wyższym będą wyświetlane razem z elementem. Jeśli wszystkie problemy zostały potwierdzone, wokół okręgu wyświetlana jest gruba zielona obwódka.

Dodatkowo:

- host w stanie maintenance jest wyróżniany pomarańczowym, wypełnionym kwadratem. Należy pamiętać, że wyróżnianie stanu maintenance ma pierwszeństwo przed wyróżnianiem ważności problemu, jeśli elementem mapy jest host.
- wyłączony host (niemonitorowany) jest wyróżniany szarym, wypełnionym kwadratem.

Wyróżnianie jest wyświetlane, jeśli pole wyboru *Wyróżnianie ikon* jest zaznaczone w konfiguracji mapy.

Wskaźniki ostatnich zmian

Czerwone trójkąty skierowane do wewnątrz, umieszczone wokół elementu, wskazują niedawną zmianę stanu wyzwalacza — taką, która nastąpiła w ciągu ostatnich 30 minut. Trójkąty te są wyświetlane, jeśli w konfiguracji mapy zaznaczono pole wyboru *Oznaczaj elementy przy zmianie stanu wyzwalacza*.

Linki

Kliknięcie elementu mapy otwiera menu z dostępnymi linkami. Kliknięcie nazwy hosta wyświetla menu hosta.

Przyciski

Przyciski po prawej stronie oferują następujące opcje:

Edit map	Przejdź do edycji zawartości mapy.

	Dodaj mapę do widżetu ulubionych w Dashboardach.
	Mapa znajduje się w widżecie ulubionych w Dashboardach. Kliknij, aby usunąć mapę z widżetu ulubionych.

Przyciski trybu widoku wspólne dla wszystkich sekcji są opisane na stronie Monitoring.

Czytelne podsumowanie na mapach

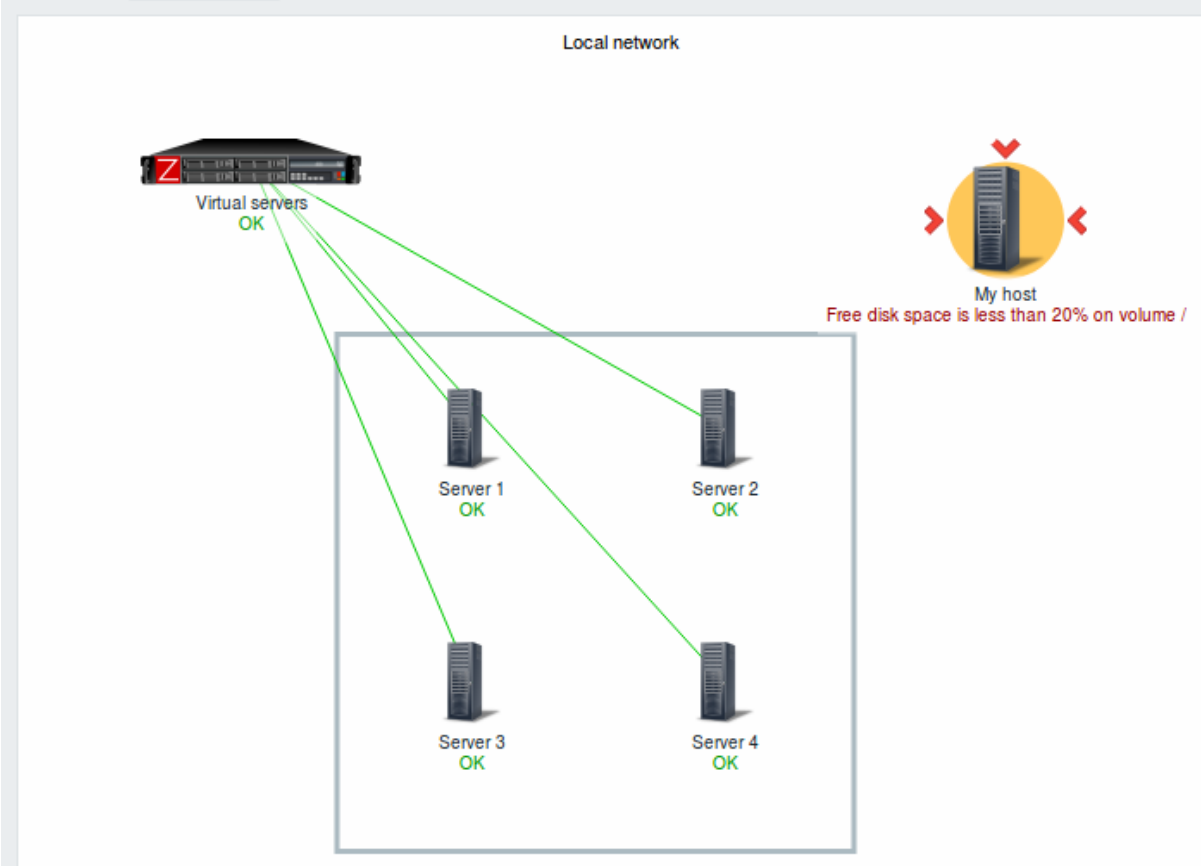
Dostępna jest ukryta właściwość „aria-label", która umożliwia odczytywanie informacji z mapy za pomocą czytnika ekranu. Dostępny jest zarówno ogólny opis mapy, jak i opis poszczególnych elementów, w następującym formacie:

- opis mapy: `<Nazwa mapy>, <* z * elementów w stanie problemu>, <* problemów łącznie>.`
- opis jednego elementu z jednym problemem: `<Typ elementu>, Status <Stan elementu>, <Nazwa elementu>, <Opis problemu>.`
- opis jednego elementu z wieloma problemami: `<Typ elementu>, Status <Stan elementu>, <Nazwa elementu>, <* problemów>.`
- opis jednego elementu bez problemów: `<Typ elementu>, Status <Stan elementu>, <Nazwa elementu>.`

Na przykład dostępny jest następujący opis:

`'Sieć lokalna, 1 z 6 elementów w stanie problemu, 1 problem łącznie. Host, Status problem, Mój host, Wolne`

dla następującej mapy:



Odwoływanie się do mapy sieciowej

Do map sieciowych można odwoływać się za pomocą parametrów GET `sysmapid` i `mapname`. Na przykład,

`http://zabbix/zabbix/zabbix.php?action=map.view&mapname=Local%20network`

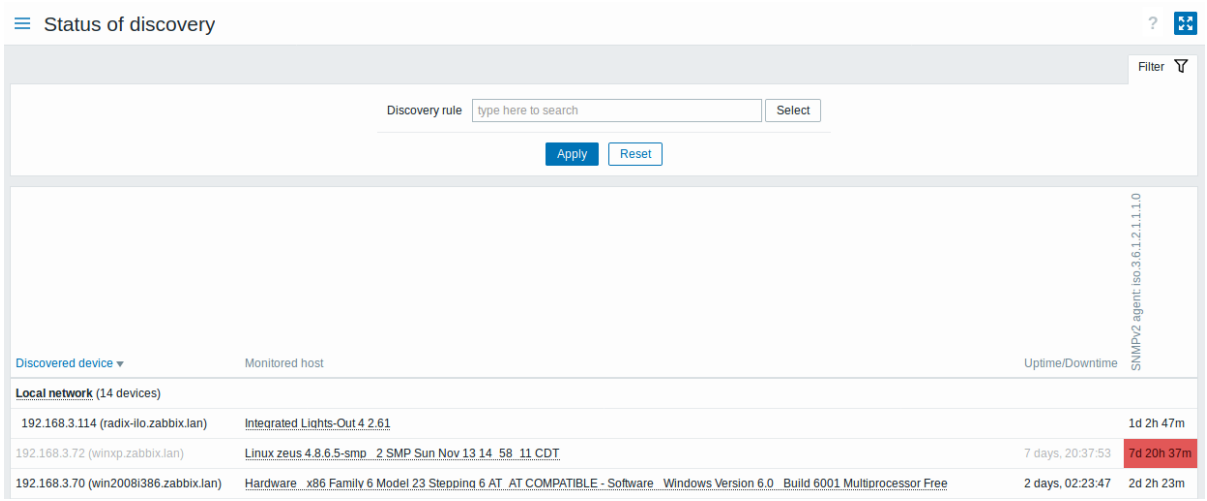
otworzy mapę o tej nazwie (Local network).

Jeśli określono zarówno `sysmapid` (ID mapy), jak i `mapname` (nazwę mapy), `mapname` ma wyższy priorytet.

Odkrywanie

Przegląd

W sekcji *Monitoring* > *Wykrywanie* wyświetlane są wyniki wykrywania sieci. Wykryte urządzenia są sortowane według reguły wykrywania.



Wyświetlane dane:

Kolumna	Opis
Wykryte urządzenie	Wykryte urządzenia są wyświetlane i grupowane według reguły wykrywania. Kliknięcie reguły wykrywania otwiera menu reguły zawierające odnośnik do formularza konfiguracji reguły wykrywania. Jeśli urządzenie ma kilka adresów IP (zgodnie z ustawieniem *Kryteria unikatowości urządzenia* reguły wykrywania), są one wyświetlane razem: najpierw główny adres IP urządzenia, a następnie pozostałe adresy IP urządzenia (z wcięciem). Głównym adresem IP jest najniższy adres IP, dla którego w kolumnie *Monitorowany host* znajduje się host, lub — jeśli nie ma takiego adresu — najniższy adres IP urządzenia. Urządzenia można sortować według ich głównego adresu IP w kolejności rosnącej lub malejącej; pozostałe adresy IP urządzenia są zawsze wyświetlane poniżej jego głównego adresu IP, w kolejności rosnącej.
Monitorowany host	Jeśli urządzenie jest już monitorowane, w tej kolumnie zostanie wyświetlona nazwa hosta. Kliknięcie nazwy hosta otwiera menu hosta.
Czas działania/przestoju	W tej kolumnie wyświetlany jest czas, przez jaki urządzenie jest wykrywane lub niedostępne od poprzedniego wykrycia. W przypadku urządzenia z kilkoma adresami IP czas jest wyświetlany tylko dla jego głównego adresu IP.
Sprawdzenie wykrywania	Wyświetlany jest stan poszczególnej usługi (sprawdzenia wykrywania) dla każdego wykrytego urządzenia. Czerwona komórka oznacza, że usługa jest niedostępna. Czas działania lub przestoju usługi jest wyświetlany w komórce. Ta kolumna jest wyświetlana tylko wtedy, gdy usługa została znaleziona na co najmniej jednym wykrytym urządzeniu.

Note:
Liczba reguł wykrywania oraz łączna liczba wykrytych urządzeń wyświetlanych w tej sekcji są ograniczone przez parametr *Limit wyników wyszukiwania i filtrowania* w sekcji *Administracja > Ogólne > GUI*. Reguły lub urządzenia przekraczające ten limit nie są wyświetlane i nie jest wyświetlana żadna informacja o jego przekroczeniu.

Przyciski

Przyciski trybu widoku wspólne dla wszystkich sekcji są opisane na stronie Monitoring.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły odkrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Gdy w filtrze nic nie jest zaznaczone, wyświetlane są wszystkie włączone reguły odkrywania. Aby wybrać konkretną regułę odkrywania do wyświetlenia, zacznij wpisywać jej nazwę w filtrze. Zostaną wyświetlone wszystkie pasujące włączone reguły odkrywania, które można wybrać. Można wybrać więcej niż jedną regułę odkrywania.

APM

Przegląd

Menu *APM* służy do obsługi funkcji monitorowania wydajności aplikacji (APM) w systemie Zabbix. Wyświetla dane telemetryczne wysyłane przez aplikacje i usługi za pomocą OpenTelemetry, otwartego standardu zbierania danych telemetrycznych. Dane te pomagają sprawdzić, jak żądania przepływają przez aplikacje i usługi, mierzyć ich wydajność oraz znajdować przyczyny powolnych lub nieudanych operacji.

Menu zawiera następujące sekcje odpowiadające typom sygnałów OpenTelemetry:

- Ślady
- Metryki
- Dzienniki

W przeciwieństwie do sekcji *Monitorowanie* > *Najnowsze dane*, która wyświetla wartości zebrane przez pozycje Zabbix, sekcje *APM* wyświetlają dane telemetryczne bezpośrednio ze źródła danych, dlatego nie trzeba tworzyć żadnych pozycji. Aby wyświetlać dane w sekcjach *APM*:

1. Określ źródło danych telemetrycznych w sekcji *Administracja* > *Źródło danych*.
2. Upewnij się, że dane telemetryczne są zapisywane w źródle danych przez proxy Zabbix z włączoną funkcją APM lub przez kolektor OpenTelemetry. Pamiętaj, że serwer Zabbix nie może odbierać danych telemetrycznych; może to robić tylko proxy Zabbix.

Aby używać danych telemetrycznych w wyzwalaczach lub na wykresach, użyj pozycji zapytanie telemetryczne.

Ślady

Przegląd

Sekcja *APM* > *Ślady* wyświetla ślady zebrane z aplikacji i usług.

Ślad przedstawia ścieżkę, którą żądanie pokonuje w aplikacjach i usługach. Na przykład, gdy użytkownik kliknie przycisk *Złóż zamówienie* w sklepie internetowym, żądanie może przejść z usługi frontend przez usługi realizacji zamówienia i płatności do bazy danych.

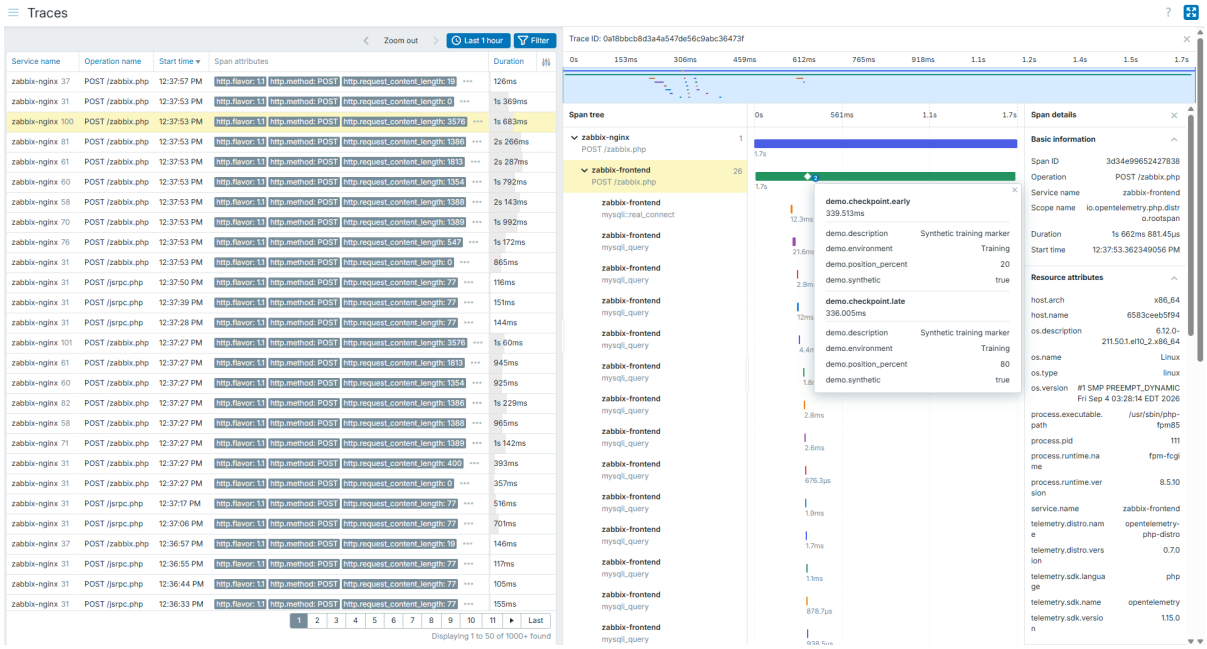
Ślad składa się z **zakresów**. Każdy zakres przedstawia pojedynczą operację, taką jak żądanie HTTP wysyłane przez usługę frontend do usługi realizacji zamówienia, zapytanie do bazy danych zapisujące zamówienie lub wywołanie funkcji obliczającej koszt wysyłki.

Zakresy tworzą hierarchię. **Zakres główny** przedstawia pierwszą operację żądania. Każdy inny zakres jest **zakresem podrzędnym** zakresu, podczas którego rozpoczęto jego operację. Na przykład usługa frontend odbierająca żądanie *Złóż zamówienie* jest zakresem głównym, a usługa realizacji zamówienia przetwarzająca zamówienie jest jego zakresem podrzędnym.

Każdy zakres jest tworzony przez **usługę** (np. `zabbix-frontend`; nazwaną za pomocą atrybutu zasobu OpenTelemetry `service.name`) oraz przez zakres instrumentacji, czyli część kodu, która go utworzyła (np. biblioteka śledząca żądania HTTP).

Aby zapewnić kontekst dotyczący każdej operacji, zakresy mają również **atrybuty** w postaci klucz-wartość: atrybuty zakresu opisują samą operację (np. `http.method: POST`), a atrybuty zasobu opisują usługę i środowisko, w którym działa (np. `host.name: eabf8c2a5ddc`).

Więcej informacji na temat śladów można znaleźć w dokumentacji OpenTelemetry.



Sekcja *APM* > *Ślady* zawiera następujące elementy:

- Przycisk Tryb kiosku - wyświetla tylko zawartość strony, bez menu i nagłówka.
- Selektor okresu - wybiera okres, dla którego mają być wyświetlane ślady.
- Filtr - wyświetla tylko interesujące Cię ślady.
- Lista śladów - umożliwia wyświetlanie śladów opisanych za pomocą ich zakresów głównych.
- Panel śladu - umożliwia wyświetlanie wszystkich zakresów wybranego śladu na osi czasu.

Lista śladów

Każdy wiersz na liście śladów reprezentuje ślad; kolumny przedstawiają dane jego głównego span.

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

Ślady są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia wyświetlanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie domyślnego układu. Można również zmienić szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślną szerokość. Wszystkie dostosowania są stosowane natychmiast i zapamiętywane dla bieżącego użytkownika.

Lista śladów może zawierać następujące kolumny:

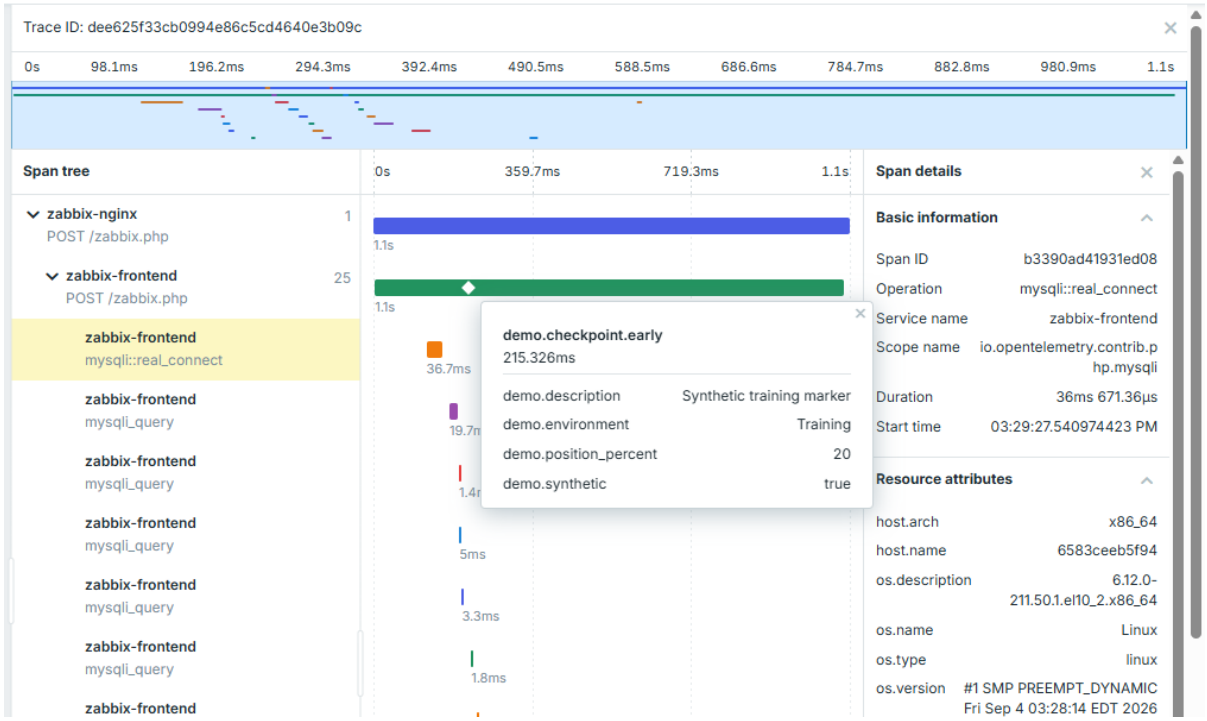
Note:
Każda kolumna odpowiada kolumnie w tabeli `otel_traces` źródła danych telemetrycznych; listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych. Większość kolumn ma taką samą nazwę jak w źródle danych (np. *Trace ID* odpowiada kolumnie `TraceId`). Jeśli nazwy się różnią, kolumna źródła danych jest określona w poniższej tabeli.

Kolumna	Opis
Trace ID	Identyfikator śladu, wspólny dla wszystkich spanów należących do tego śladu.
Span ID	Identyfikator głównego span.

Kolumna	Opis
Trace state	Dodatkowe dane, jeśli istnieją, dołączane do śladu przez narzędzia śledzenia (zobacz W3C Trace Context).
Service name	Usługa, która utworzyła główny span, a następnie liczba spanów w śladzie. Jeśli którykolwiek span ma kod stanu *Error* (zobacz poniżej), czerwona ikona pokazuje ich liczbę.
Operation name	Operacja reprezentowana przez główny span (np. `POST /zabbix.php`). Kolumna źródła danych: `SpanName`.
Span kind	Rola głównego span: **Unspecified** - nieokreślona. **Internal** - operacja wykonywana wewnątrz aplikacji. **Server** - obsługa przychodzącego żądania. **Client** - żądanie wychodzące. **Producer** - utworzenie operacji asynchronicznej (np. wysłanie wiadomości). **Consumer** - przetwarzanie operacji asynchronicznej.
Scope name	Zakres instrumentacji, który utworzył główny span.
Scope version	Wersja zakresu instrumentacji.
Status code	Stan operacji reprezentowanej przez główny span: **Unset** - głównemu spanowi nie przypisano jawnego stanu. **Ok** - operacja została pomyślnie zakończona. **Error** - operacja została zakończona błędem.
Status message	Dodatkowe informacje, jeśli istnieją, dotyczące stanu.
Start time	Data i godzina rozpoczęcia głównego span. Kolumna źródła danych: `Timestamp`.
Span attributes	Atrybuty span głównego span. Kliknij przycisk z trzema kropkami ⋯, aby wyświetlić wszystkie atrybuty.
Resource attributes	Atrybuty zasobu głównego span. Kliknij przycisk z trzema kropkami ⋯, aby wyświetlić wszystkie atrybuty.
Duration	Czas trwania śladu; szary pasek pokazuje go w odniesieniu do pozostałych śladów.

Panel śladu

Kliknij ślad na liście śladów, aby otworzyć panel śladu, który pokazuje wszystkie zakresy śladu na osi czasu: jakie operacje zostały wykonane, w jakiej kolejności i ile czasu zajęła każda z nich.



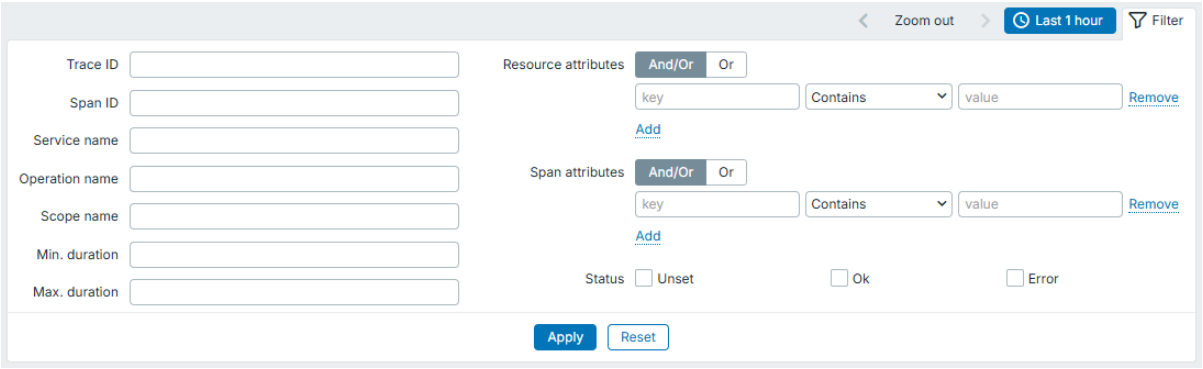
Panel śladu może zawierać następujące elementy:

Element	Opis
Trace ID	Identyfikator wybranego śladu.

Element	Opis
Przegląd osi czasu śladu	Cały ślad z zakresami w maksymalnie 8 wierszach (kilka zakresów może współdzielić jeden wiersz). Przeciągnij krawędzie wybranego zakresu lub sam zakres, aby powiększyć oś czasu zakresów dla tej części śladu.
Drzewo zakresów	Zakresy przedstawione hierarchicznie; każdy pokazuje swoją usługę (pogrubioną czcionką), nazwę operacji i liczbę bezpośrednich zakresów podrzędnych.
Oś czasu zakresów	Zakresy przedstawione jako paski: położenie i długość pokazują, kiedy rozpoczął się każdy zakres i ile czasu trwał. Znaczniki pokazują zdarzenia (np. wyjątki); najedź kursorem na znacznik lub kliknij go, aby wyświetlić szczegóły.
Szczegóły zakresu	Kliknij zakres w drzewie zakresów, aby wyświetlić jego szczegóły i atrybuty.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlać tylko interesujące Cię ślady.



Aby filtrować według atrybutów, wprowadź klucz atrybutu, wybierz operator i wprowadź wartość atrybutu.

Obsługiwane są następujące operatory:

- **Zawiera** - *(domyślnie)* wartość atrybutu zawiera wprowadzony ciąg znaków.
- **Równa się** - wartość atrybutu jest równa wprowadzonemu ciągowi znaków.
- **Nie zawiera** - wartość atrybutu nie zawiera wprowadzonego ciągu znaków.
- **Nie równa się** - wartość atrybutu nie jest równa wprowadzonemu ciągowi znaków.
- **Istnieje** - atrybut z wprowadzonym kluczem istnieje.
- **Nie istnieje** - atrybut z wprowadzonym kluczem nie istnieje.

Jeśli dodasz kilka warunków, wybierz sposób ich oceny:

- **And/Or** - *(domyślnie)* wszystkie warunki muszą być spełnione; warunki z tym samym kluczem atrybutu są łączone operatorem *Or*.
- **Or** - musi być spełniony co najmniej jeden warunek.

Metryki

Przegląd

Sekcja *APM* > *Metrics* wyświetla punkty danych metryk zebrane z aplikacji i usług.

Metryka to pomiar (np. czas trwania żądania, liczba przetworzonych zamówień lub użycie pamięci). Aplikacje wysyłają swoje wartości jako **punkty danych**, które zawierają wartość (lub zagregowane wartości) zmierzoną w określonym czasie. Metryka jest podobna do pozycji Zabbix, a punkt danych jest podobny do wartości tej pozycji.

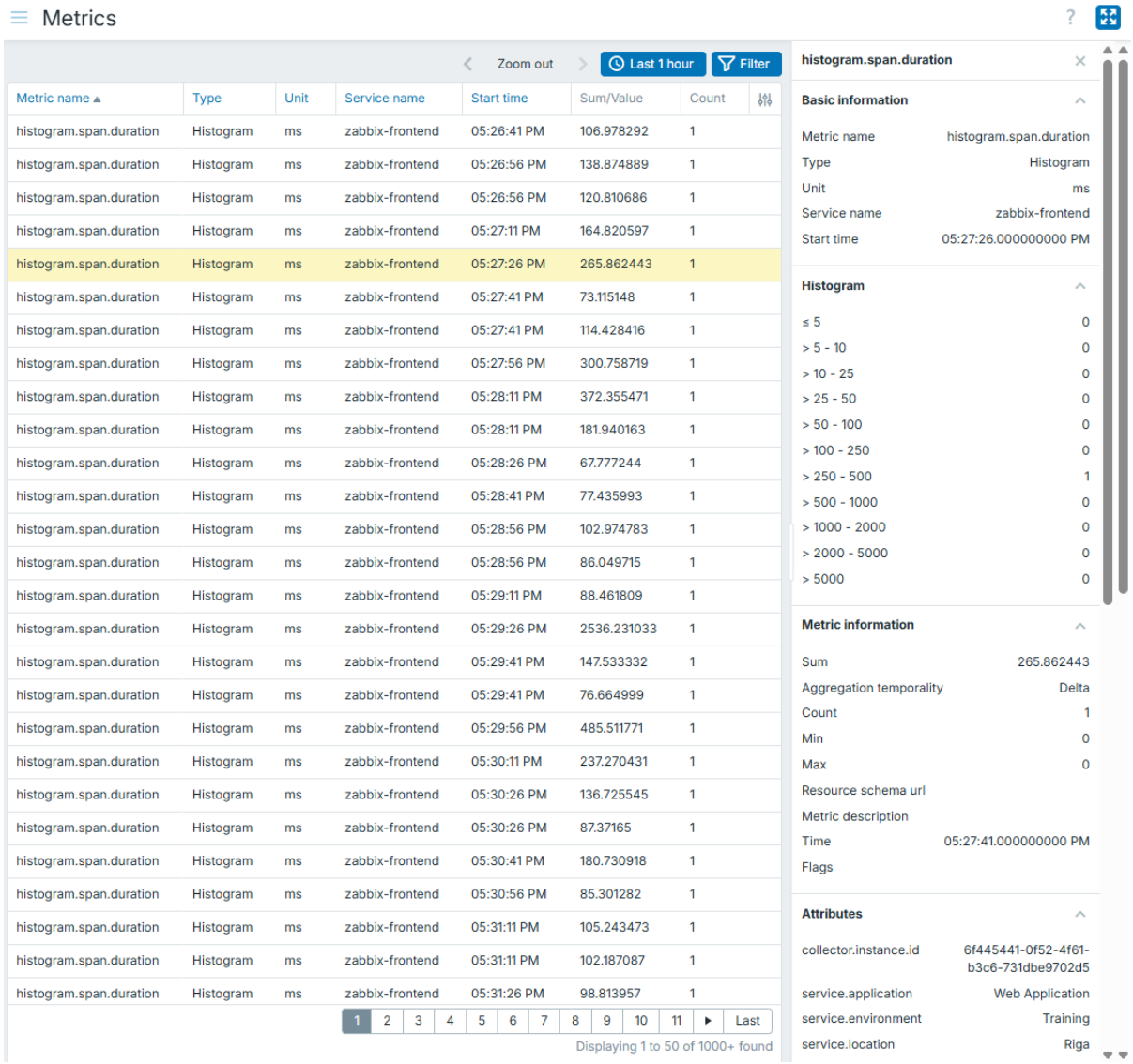
Punkt danych może być jednego z następujących typów:

- **Gauge** - wartość zmierzona w określonym czasie (np. bieżące użycie pamięci).
- **Sum** - suma pomiarów w określonym przedziale czasu (np. łączna liczba żądań).
- **Histogram** - liczba pomiarów w określonym przedziale czasu, które mieszczą się w każdym z kilku ustalonych zakresów wartości, nazywanych kubełkami (np. ile żądań trwało 0-100 ms, 100-250 ms lub 250-500 ms).
- **Histogram wykładniczy** - podobnie jak histogram, ale każdy kubełek jest szerszy od poprzedniego o ten sam współczynnik (np. 1-2 ms, 2-4 ms, 4-8 ms).

Każdy punkt danych jest tworzony przez **usługę** (np. `zabbix-frontend`; nazwa jest określana przez atrybut zasobu OpenTelemetry `service.name`) oraz przez zakres instrumentacji, czyli część kodu, która go utworzyła (np. biblioteka mierząca czas trwania żądań HTTP).

Aby zapewnić kontekst dla każdego pomiaru, punkty danych mają również **atrybuty** klucz-wartość: atrybuty punktu danych opisują, co zostało zmierzone, i rozróżniają punkty danych tej samej metryki (np. `http.request.method: POST`), natomiast atrybuty zasobu opisują usługę i środowisko, w którym działa (np. `host.name: eabf8c2a5ddc`).

Więcej informacji na temat metryk można znaleźć w dokumentacji OpenTelemetry.

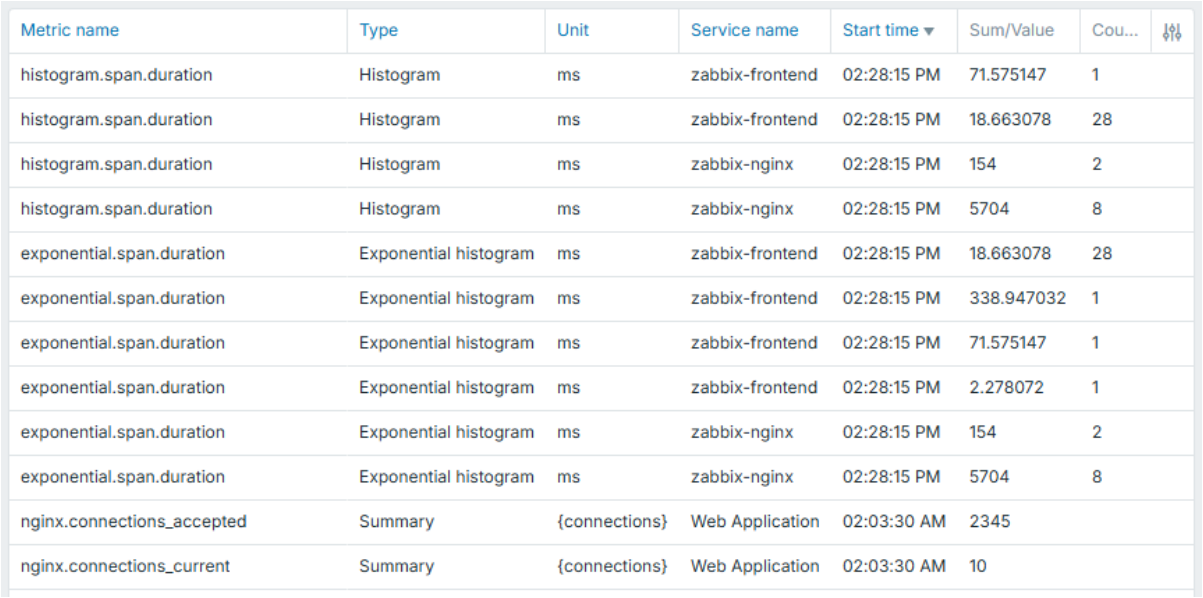


Sekcja *APM* > *Metrics* zawiera następujące elementy:

- Przycisk trybu kiosku - wyświetla tylko zawartość strony, bez menu i nagłówka.
- Selektor przedziału czasu - wybór przedziału czasu, dla którego mają być wyświetlane punkty danych.
- Filtr - wyświetla tylko interesujące Cię punkty danych.
- Lista metryk - wyświetlanie punktów danych.
- Szczegóły metryki - wyświetlanie szczegółów wybranego punktu danych.

Lista metryk

Każdy wiersz na liście metryk reprezentuje punkt danych.



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	

Punkty danych są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia wyświetlanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie domyślnego układu. Można również zmienić szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślną szerokość. Wszystkie dostosowania są stosowane natychmiast i zapamiętywane dla bieżącego użytkownika.

Lista metryk może zawierać następujące kolumny:

Note:
Każda kolumna, z wyjątkiem *Type*, odpowiada kolumnie w tabelach metryk (`otel_metrics_gauge`, `otel_metrics_sum`, `otel_metrics_histogram` i `otel_metrics_exponential_histogram`) źródła danych telemetrycznych; listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych. Większość kolumn ma taką samą nazwę jak w źródle danych (np. *Metric name* odpowiada kolumnie `MetricName`). Jeśli nazwy się różnią, nazwa kolumny źródła danych jest podana w poniższej tabeli.

Column	Description
Metric name	Metryka, do której należy punkt danych (np. `http.server.duration`).
Type	Typ punktu danych: *Gauge*, *Sum*, *Histogram* lub *Exponential histogram*. Określany na podstawie tabeli metryk zawierającej punkt danych.
Unit	Jednostka, jeśli została określona, wartości metryki (np. `s` lub `By`; zobacz UCUM). Kolumna źródła danych: `MetricUnit`.
Service name	Usługa, która utworzyła punkt danych.
Scope name	Zakres instrumentacji, który utworzył punkt danych.
Start time	Data i czas rozpoczęcia okresu pomiarowego punktu danych. Kolumna źródła danych: `StartTimeUnix`.
Sum/Value	Suma wszystkich pomiarów (*Histogram*, *Exponential histogram*) lub wartość (*Gauge*, *Sum*) punktu danych.
Count	Liczba pomiarów w punkcie danych typu *Histogram* lub *Exponential histogram*.
Time	Data i czas punktu danych; w przypadku punktów danych typu *Sum*, *Histogram* i *Exponential histogram* koniec okresu pomiarowego. Kolumna źródła danych: `TimeUnix`.
Flags	Flagi punktu danych: **0** - nie ustawiono żadnych flag. **1** - dla punktu danych nie zarejestrowano żadnej wartości.
Aggregation temporality	Sposób agregowania w czasie wartości punktu danych typu *Sum*, *Histogram* lub *Exponential histogram*: **Unspecified** - nie określono. **Delta** - agregowane w okresie od poprzedniego punktu danych. **Cumulative** - agregowane w okresie od czasu rozpoczęcia.
Metric description	Opis metryki, jeśli został określony.
Attributes	Atrybuty punktu danych. Kliknij przycisk z trzema kropkami ••• , aby wyświetlić wszystkie atrybuty.
Resource attributes	Atrybuty zasobu punktu danych. Kliknij przycisk z trzema kropkami ••• , aby wyświetlić wszystkie atrybuty.

Column	Description
Scope attributes	Atrybuty zakresu instrumentacji. Kliknij przycisk z trzema kropkami ⋯ , aby wyświetlić wszystkie atrybuty.

Szczegóły metryki

Kliknij punkt danych na liście metryk, aby otworzyć panel szczegółów metryki, w którym wyświetlane są wszystkie informacje o punkcie danych.

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

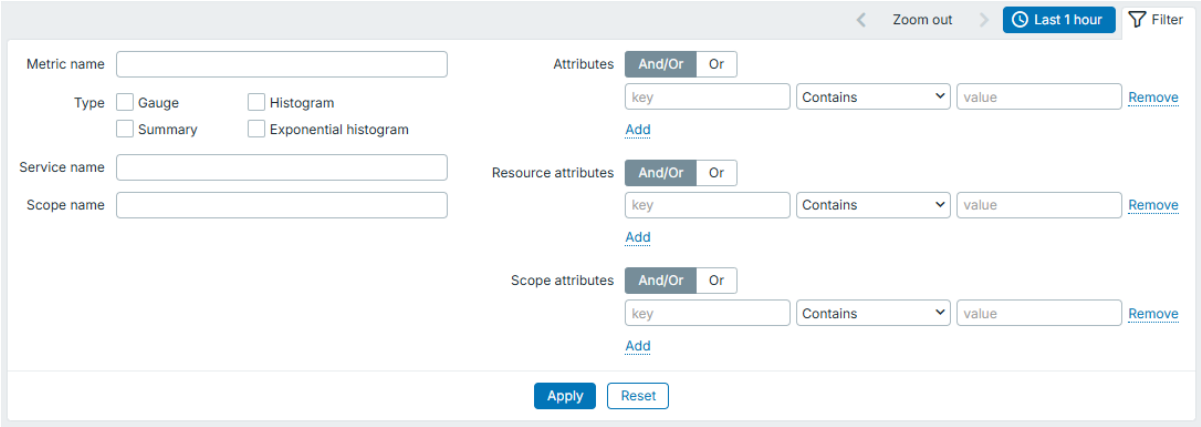
Scope

Panel szczegółów metryki może zawierać następujące sekcje:

Sekcja	Opis
Podstawowe informacje	Nazwa metryki, typ, jednostka, nazwa usługi i czas rozpoczęcia.
Histogram	Przedziały (np. `> 5 – 10`), z których każdy zawiera liczbę pomiarów w danym przedziale. Wyświetlane tylko dla punktów danych typu *Histogram*.
Histogram wykładniczy	Przedziały (np. `>= 249.7885 – 249.7887`), z których każdy zawiera liczbę pomiarów w danym przedziale. Wyświetlane tylko dla punktów danych typu *Histogram wykładniczy*.
Informacje o metryce	Wartości punktu danych, zależne od jego typu, a także opis metryki, czas, flagi i adres URL schematu zasobu.
Atrybuty	Atrybuty punktu danych.
Atrybuty zasobu	Atrybuty zasobu.
Atrybuty zakresu	Atrybuty zakresu instrumentacji, jeśli istnieją.
Zakres	Nazwa, wersja i adres URL schematu zakresu instrumentacji.

Korzystanie z filtra

Filtr umożliwia wyświetlanie tylko interesujących Cię punktów danych.



Aby filtrować według atrybutów, wprowadź klucz atrybutu, wybierz operator i wprowadź wartość atrybutu.

Obsługiwane są następujące operatory:

- **Zawiera** - *(domyślnie)* wartość atrybutu zawiera wprowadzony ciąg znaków.
- **Równa się** - wartość atrybutu jest równa wprowadzonemu ciągowi znaków.
- **Nie zawiera** - wartość atrybutu nie zawiera wprowadzonego ciągu znaków.
- **Nie równa się** - wartość atrybutu nie jest równa wprowadzonemu ciągowi znaków.
- **Istnieje** - atrybut o wprowadzonym kluczu istnieje.
- **Nie istnieje** - atrybut o wprowadzonym kluczu nie istnieje.

Jeśli dodasz kilka warunków, wybierz sposób ich oceny:

- **And/Or** - *(domyślnie)* wszystkie warunki muszą być spełnione; warunki z tym samym kluczem atrybutu są łączone operatorem *Or*.
- **Or** - musi być spełniony co najmniej jeden warunek.

Dzienniki

Przegląd

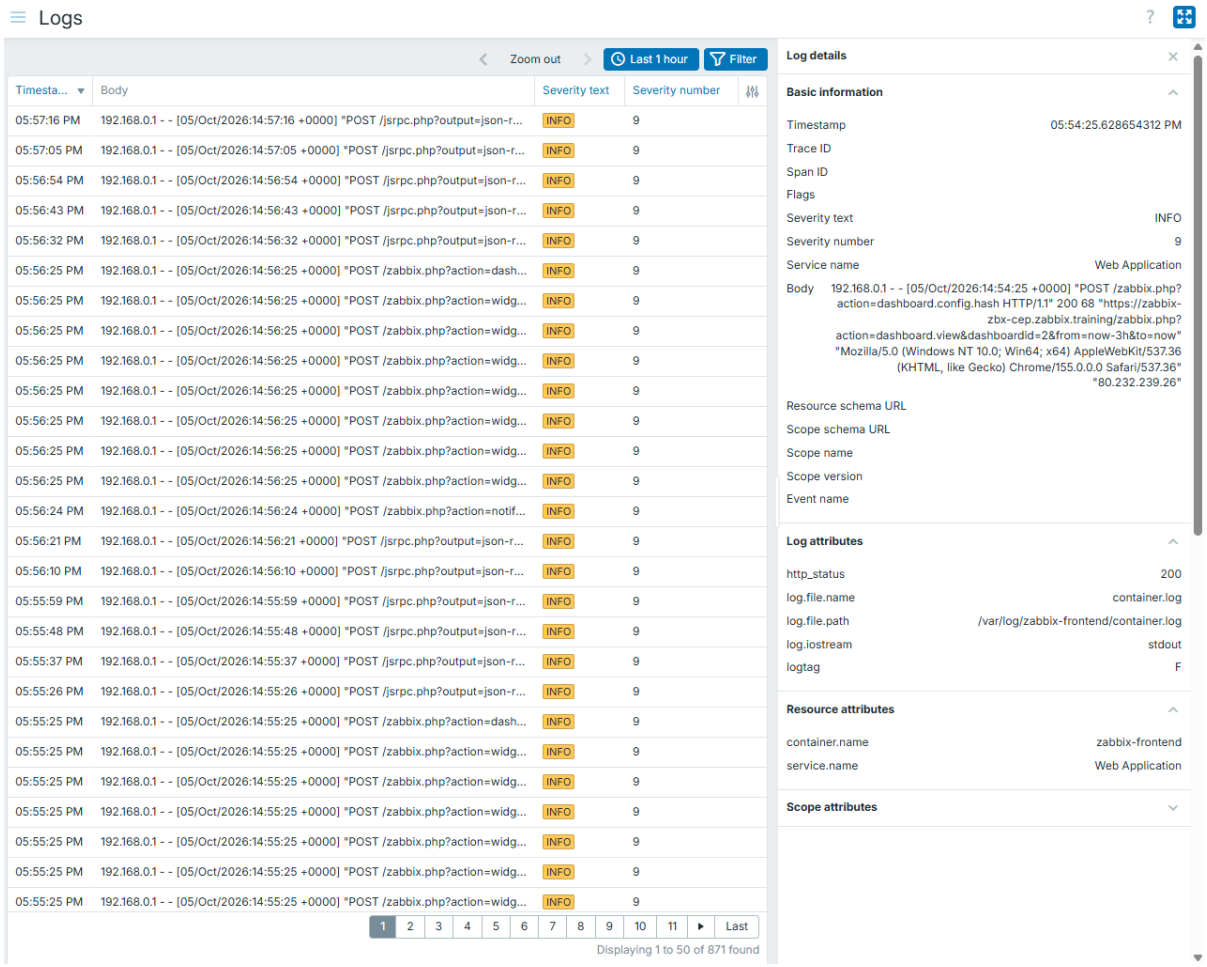
Sekcja *APM* > *Dzienniki* wyświetla rekordy dziennika zebrane z aplikacji i usług.

Rekord dziennika to tekstowy zapis zdarzenia (np. nieudanej płatności lub przekroczenia limitu czasu połączenia z bazą danych). Każdy rekord dziennika ma poziom ważności i komunikat (treść). Jeśli rekord dziennika został utworzony podczas operacji będącej częścią śladu, zawiera również identyfikatory tego śladu i odcinka.

Każdy rekord dziennika jest tworzony przez **usługę** (np. `zabbix-frontend`; nazwa jest określana przez atrybut zasobu OpenTelemetry `service.name`) oraz przez zakres instrumentacji, czyli część kodu, która go utworzyła (np. bibliotekę rejestrowania).

Aby zapewnić kontekst każdego zdarzenia, rekordy dziennika zawierają również **atrybuty** klucz-wartość: atrybuty dziennika opisują samo zdarzenie (np. `log.file.path: /var/log/zabbix/zabbix_frontend.log`), a atrybuty zasobu opisują usługę i środowisko, w którym działa (np. `host.name: eabf8c2a5ddc`).

Więcej informacji na temat dzienników można znaleźć w dokumentacji OpenTelemetry.

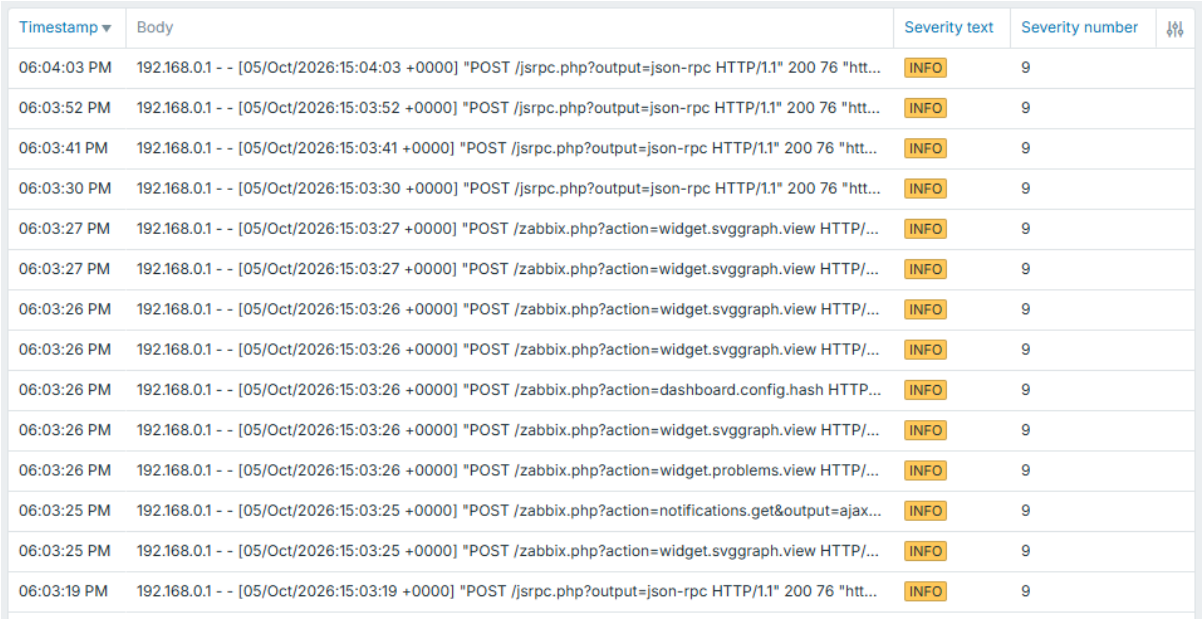


Sekcja *APM* > *Dzienniki* zawiera następujące elementy:

- Przycisk Tryb kiosku - wyświetla tylko zawartość strony, bez menu i nagłówka.
- Selektor okresu - wybór okresu, dla którego mają być wyświetlane rekordy dziennika.
- Filtr - wyświetla tylko interesujące użytkownika rekordy dziennika.
- Lista dzienników - wyświetlanie rekordów dziennika.
- Szczegóły dziennika - wyświetlanie szczegółów wybranego rekordu dziennika.

Lista dzienników

Każdy wiersz na liście dzienników reprezentuje rekord dziennika.



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

Rekordy dziennika są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia wyświetlanie, ukrywanie i zmienianie kolejności kolumn oraz przywrócenie domyślnego układu. Można również zmienić szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślną szerokość. Wszystkie dostosowania są stosowane natychmiast i zapamiętywane dla bieżącego użytkownika.

Lista dzienników może zawierać następujące kolumny:

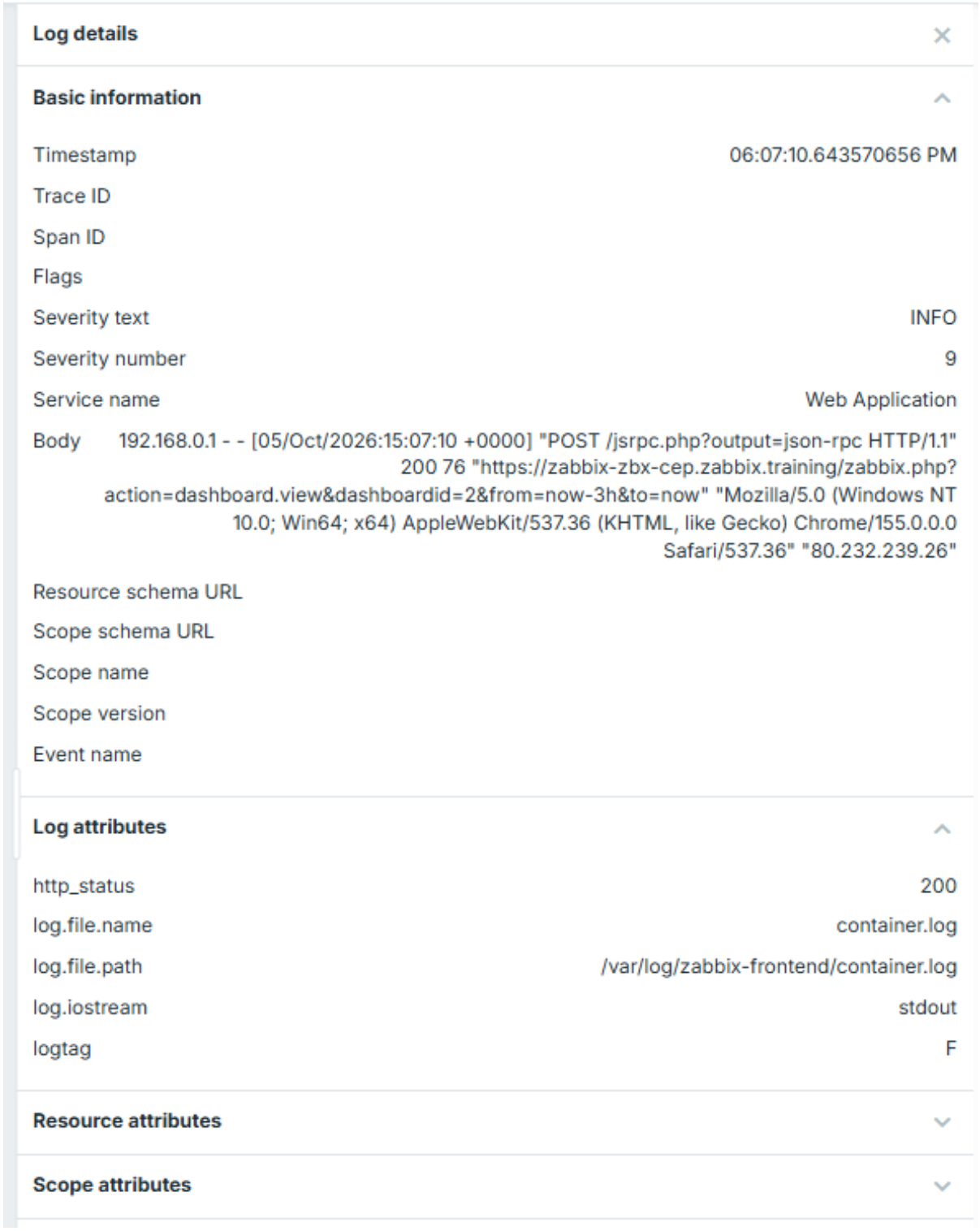
Note:
Każda kolumna odpowiada kolumnie w tabeli `otel_logs` źródła danych telemetrycznych; listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych. Większość kolumn ma taką samą nazwę jak w źródle danych (np. *Tekst ważności* odpowiada kolumnie `SeverityText`). Jeśli nazwy się różnią, kolumna źródła danych jest określona w poniższej tabeli.

Kolumna	Opis
Znacznik czasu	Data i godzina wystąpienia zdarzenia.
Treść	Komunikat rekordu dziennika.
Tekst ważności	Poziom ważności podany przez aplikację (np. `INFO`, `WARN` lub `ERROR`). Kolor etykiety zależy od ważności.
Numer ważności	Poziom ważności jako liczba: **0** - UNSPECIFIED. **1-4** - TRACE. **5-8** - DEBUG. **9-12** - INFO. **13-16** - WARN. **17-20** - ERROR. **21-24** - FATAL.
ID śladu	ID, jeśli istnieje, śladu, podczas którego utworzono rekord dziennika.
ID zakresu	ID, jeśli istnieje, zakresu, podczas którego utworzono rekord dziennika.
Flagi	Flagi śladu rekordu dziennika: **0** - ślad nie jest próbkowany. **1** - ślad jest próbkowany (rejestrowany). Kolumna źródła danych: `TraceFlags`.
Nazwa usługi	Usługa, która utworzyła rekord dziennika.
Nazwa zakresu	Zakres instrumentacji, który utworzył rekord dziennika.
Wersja zakresu	Wersja zakresu instrumentacji.
Nazwa zdarzenia	Nazwa, jeśli istnieje, typu zdarzenia reprezentowanego przez rekord dziennika.
Atrybuty dziennika	Atrybuty dziennika rekordu dziennika. Kliknij przycisk z trzema kropkami ••• , aby wyświetlić wszystkie atrybuty.
Atrybuty zasobu	Atrybuty zasobu rekordu dziennika. Kliknij przycisk z trzema kropkami ••• , aby wyświetlić wszystkie atrybuty.

Kolumna	Opis
Atrybuty zakresu	Atrybuty zakresu instrumentacji. Kliknij przycisk z trzema kropkami ••• , aby wyświetlić wszystkie atrybuty.

Szczegóły dziennika

Kliknij rekord dziennika na liście dzienników, aby otworzyć panel *Szczegóły dziennika*, który wyświetla wszystkie informacje o rekordzie dziennika.



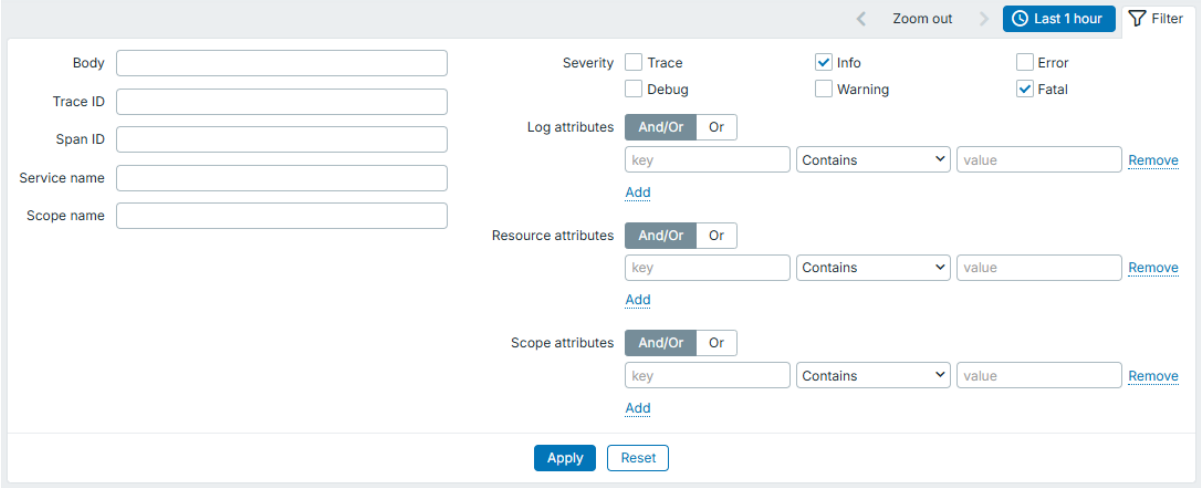
Panel *Szczegóły dziennika* może zawierać następujące sekcje:

Sekcja	Opis
Podstawowe informacje	Identyfikator śledzenia, identyfikator zakresu i flagi śledzenia rekordu dziennika.
Atrybuty dziennika	Atrybuty dziennika rekordu dziennika.

Sekcja	Opis
Atrybuty zasobu	Adres URL schematu i atrybuty zasobu.
Atrybuty zakresu	Adres URL schematu, nazwa, wersja i atrybuty zakresu instrumentacji.

Korzystanie z filtra

Filtr umożliwia wyświetlanie tylko interesujących Cię rekordów dziennika.



Aby filtrować według atrybutów, wprowadź klucz atrybutu, wybierz operator i wprowadź wartość atrybutu.

Obsługiwane są następujące operatory:

- **Zawiera** - *(domyślnie)* wartość atrybutu zawiera wprowadzony ciąg znaków.
- **Równa się** - wartość atrybutu jest równa wprowadzonemu ciągowi znaków.
- **Nie zawiera** - wartość atrybutu nie zawiera wprowadzonego ciągu znaków.
- **Nie równa się** - wartość atrybutu nie jest równa wprowadzonemu ciągowi znaków.
- **Istnieje** - atrybut z wprowadzonym kluczem istnieje.
- **Nie istnieje** - atrybut z wprowadzonym kluczem nie istnieje.

Jeśli dodasz kilka warunków, wybierz sposób ich oceny:

- **And/Or** - *(domyślnie)* wszystkie warunki muszą być spełnione; warunki z tym samym kluczem atrybutu są łączone za pomocą *Or*.
- **Or** - musi być spełniony co najmniej jeden warunek.

Usługi

Przegląd

Menu Usługi służy do funkcji monitorowania usług w Zabbix.

Usługi

Przegląd

W tej sekcji możesz zobaczyć ogólny stan wszystkich usług skonfigurowanych w Zabbixie, na podstawie Twojej infrastruktury.

Usługa może być hierarchią składającą się z kilku poziomów innych usług, nazywanych usługami „podrzędnymi”, które wpływają na ogólny stan usługi (zobacz także przegląd funkcji monitorowania usług).

Główne kategorie stanu usługi to *OK* lub *Problem*, przy czym stan *Problem* jest wyrażany za pomocą odpowiedniej nazwy i koloru ważności problemu.

Tryb wyświetlania umożliwia monitorowanie usług wraz z ich stanem i innymi szczegółami. W tej sekcji możesz również konfigurować hierarchię usług (dodawać/edytować usługi i usługi podrzędne), przełączając się do trybu edycji.

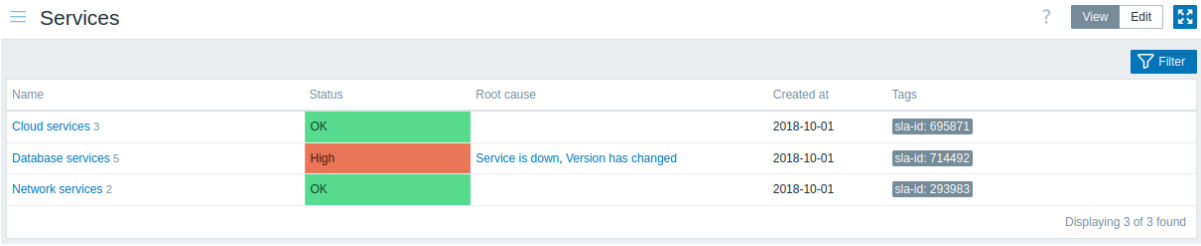
Aby przełączyć się z trybu wyświetlania do trybu edycji (i odwrotnie), kliknij odpowiedni przycisk w prawym górnym rogu:

- View - wyświetlanie usług

- Edit - dodawanie/edytowanie usług i usług podrzędnych

Pamiętaj, że dostęp do edycji zależy od ustawień roli użytkownika.

Wyświetlanie usług



Wyświetlana jest lista istniejących usług.

Wyświetlane dane:

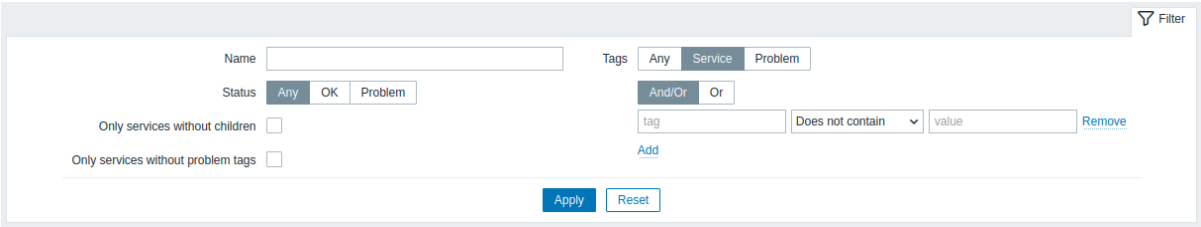
Parameter	Description
Name	Nazwa usługi. Nazwa usługi jest łączem do szczegółów usługi. Liczba po nazwie wskazuje, ile usług podrzędnych ma dana usługa.
Status	Stan usługi: **OK** - brak problemów **<problem color and severity>** - wskazuje problem i jego ważność. W przypadku wielu problemów wyświetlane są kolor i ważność najbardziej krytycznego problemu.
Root cause	Wyświetlane są problemy źródłowe, które bezpośrednio lub pośrednio wpływają na stan usługi. Te same problemy są wyświetlane zgodnie z wynikiem makra {SERVICE.ROOTCAUSE} macro. Kliknij nazwę problemu, aby zobaczyć więcej szczegółów w *Monitoring* > *Problems*. Problemy, które nie wpływają na stan usługi, nie są uwzględniane na liście.
Created at	Wyświetlany jest czas utworzenia usługi.
Tags	Wyświetlane są tagi usługi. Tagi służą do identyfikowania usługi w akcjach usługi oraz w SLA.

Przyciski

Przyciski trybu widoku, wspólne dla wszystkich sekcji, zostały opisane na stronie Monitoring.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te usługi, które Cię interesują.



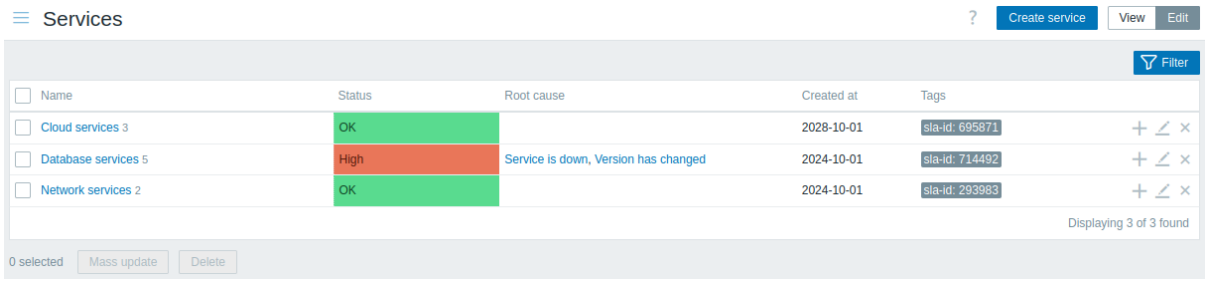
Parametr	Opis
Nazwa	Filtruj według nazwy usługi.
Status	Filtruj według stanu usługi.
Tylko usługi bez usług podrzędnych	Zaznacz pole wyboru, aby wyświetlić tylko usługi bez usług podrzędnych. Ten parametr jest dostępny tylko w trybie edycji usługi.
Tylko usługi bez tagów problemów	Zaznacz pole wyboru, aby wyświetlić tylko usługi bez tagów problemów. Ten parametr jest dostępny tylko w trybie edycji usługi.

Parametr	Opis
Tagi	Filtruj według nazwy i wartości tagu usługi albo nazwy i wartości tagu problemu usługi (w trybie edycji usługi). Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów i wartości tagów. Można ustawić kilka warunków. Dopasowanie nazw tagów zawsze uwzględnia wielkość liter. Dla każdego warunku dostępnych jest kilka operatorów: **Istnieje** - uwzględnij określone nazwy tagów; **Równa się** - uwzględnij określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Zawiera** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter); **Nie istnieje** - wyklucz określone nazwy tagów; **Nie równa się** - wyklucz określone nazwy i wartości tagów (z uwzględnieniem wielkości liter); **Nie zawiera** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez uwzględniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki mające tę samą nazwę tagu będą grupowane operatorem Or; **Or** - wystarczy, że jeden warunek zostanie spełniony.

Edycja usług

Kliknij przycisk *Edit*, aby przejść do trybu edycji. W trybie edycji lista zostaje uzupełniona o pola wyboru przed wpisami oraz o następujące dodatkowe opcje:

-  - dodaj usługę podrzędną do tej usługi
-  - edytuj tę usługę
-  - usuń tę usługę

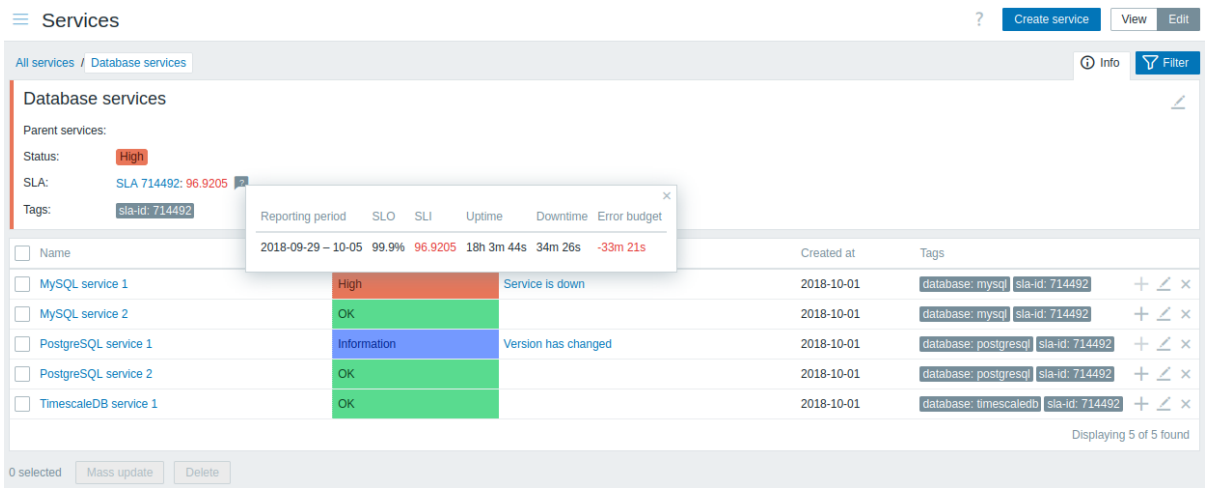


Aby skonfigurować nową usługę, kliknij przycisk *Create service* w prawym górnym rogu.

Szczegóły usługi

Aby uzyskać dostęp do szczegółów usługi, kliknij nazwę usługi. Aby wrócić do listy wszystkich usług, kliknij *All services*.

Szczegóły usługi obejmują pole informacji oraz listę usług podrzędnych.



Aby uzyskać dostęp do pola informacji, kliknij kartę *Info*. Pole informacji zawiera następujące wpisy:

- Nazwy usług nadrzędnych (jeśli istnieją)
- Bieżący status tej usługi
- Bieżące SLA tej usługi, w formacie `nazwa SLA:wskaźnik poziomu usługi`. „nazwa SLA" jest również linkiem do raportu SLA dla tej usługi. Jeśli umieścisz kursor myszy na polu informacji obok wskaźnika poziomu usługi (SLI), zostanie wyświetlona wyskakująca lista informacji ze szczegółami SLI. Wskaźnik poziomu usługi pokazuje bieżący poziom usługi w procentach.
- Tagi usługi

Pole informacji zawiera również link do konfiguracji usługi.

Aby użyć filtra dla usług podrzędnych, kliknij kartę *Filter*.

W trybie edycji lista usług podrzędnych jest uzupełniona o dodatkowe opcje edycji:

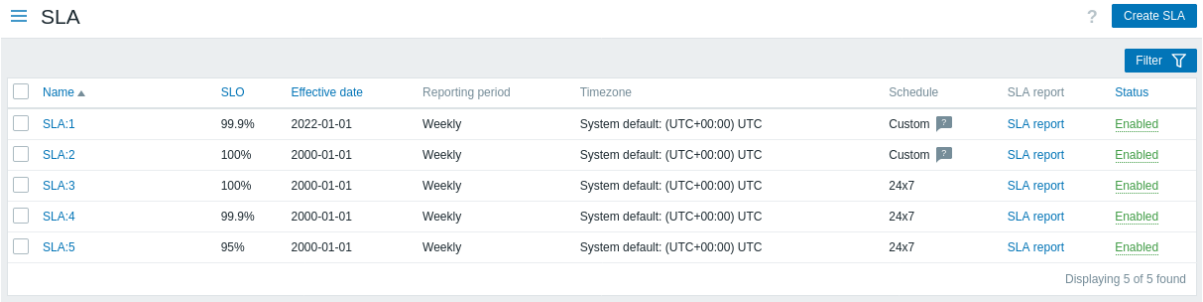
-  - dodaj usługę podrzędną do tej usługi
-  - edytuj tę usługę
-  - usuń tę usługę

SLA

Przegląd

Ta sekcja umożliwia wyświetlanie i konfigurowanie umów SLA.

SLA



Wyświetlana jest lista skonfigurowanych SLA. *Uwaga*: wyświetlane będą tylko SLA powiązane z usługami dostępnymi dla użytkownika (w trybie tylko do odczytu, chyba że dla roli użytkownika włączono opcję *Manage SLA*).

Wyświetlane dane:

Parameter	Description
Name	Wyświetlana jest nazwa SLA. Nazwa jest linkiem do konfiguracji SLA.
SLO	Wyświetlany jest cel poziomu usług (SLO).
Effective date	Wyświetlana jest data rozpoczęcia obliczania SLA.
Reporting period	Wyświetlany jest okres używany w raporcie SLA - *daily*, *weekly*, *monthly*, *quarterly* lub *annually*.
Time zone	Wyświetlana jest strefa czasowa SLA.
Schedule	Wyświetlany jest harmonogram SLA - 24x7 lub niestandardowy.
SLA report	Kliknij link, aby wyświetlić raport SLA dla tego SLA.
Status	Wyświetlany jest status SLA - włączone lub wyłączone.

Raport SLA

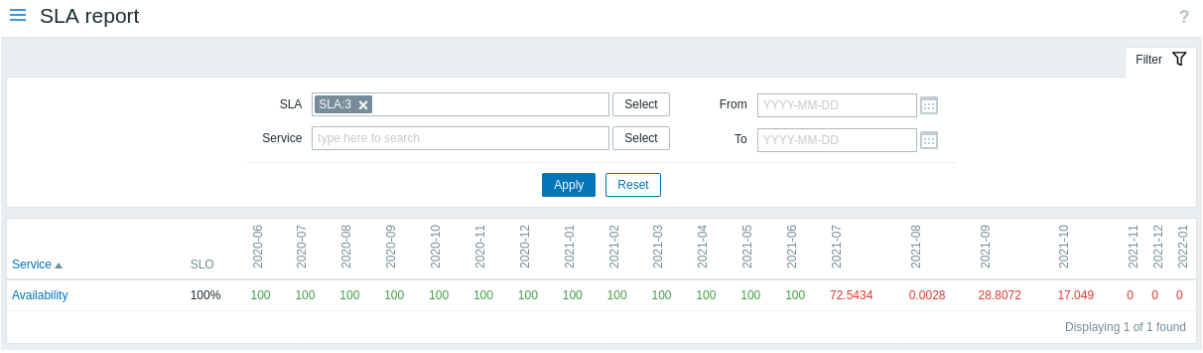
Przegląd

Ta sekcja umożliwia wyświetlanie raportów SLA na podstawie kryteriów wybranych w filtrze.

Raporty SLA można również wyświetlać jako widżet pulpitu nawigacyjnego.

Raport

Filtr umożliwia wybranie raportu na podstawie nazwy SLA, a także nazwy usługi. Możliwe jest również ograniczenie wyświetlanego okresu.

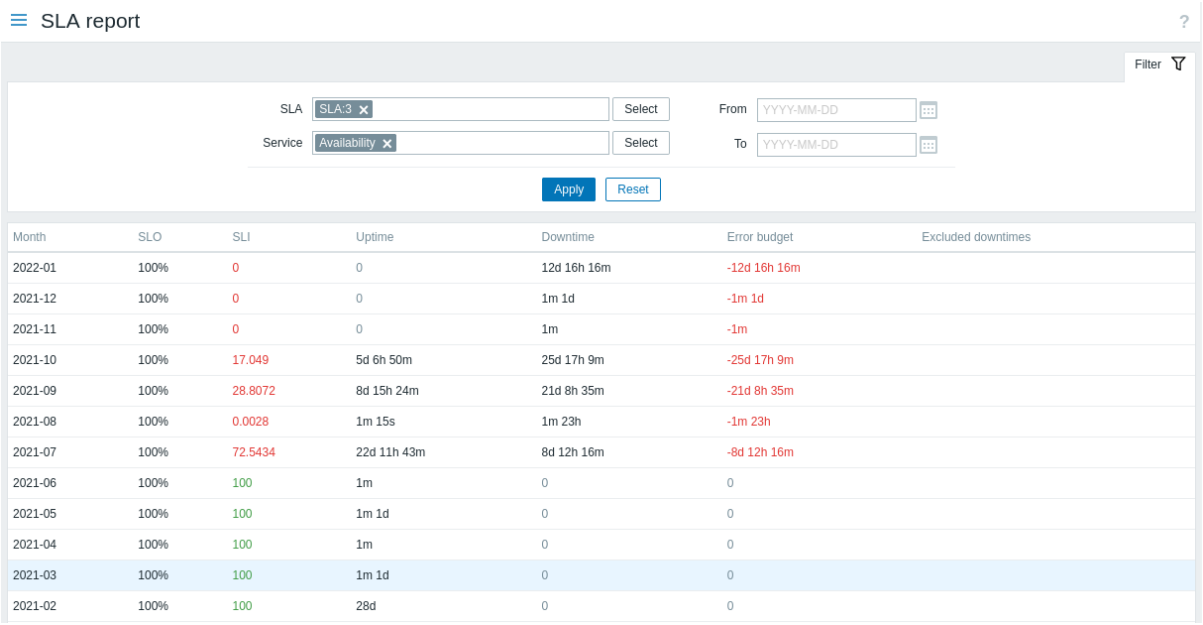


Każda kolumna (okres) wyświetla SLI dla danego okresu. Wskaźniki SLI, które naruszają ustawiony SLO, są wyróżnione na czerwono.

W raporcie wyświetlanych jest 20 okresów. Można wyświetlić maksymalnie 100 okresów, jeśli określono zarówno datę *Od*, jak i datę *Do*.

Szczegóły raportu

Jeśli klikniesz nazwę usługi w raporcie, możesz uzyskać dostęp do innego raportu, który wyświetla bardziej szczegółowy widok.



Należy pamiętać, że ujemny czas trwania problemu nie wpływa na obliczanie SLA ani na raportowanie.

Inwentarz

Przegląd

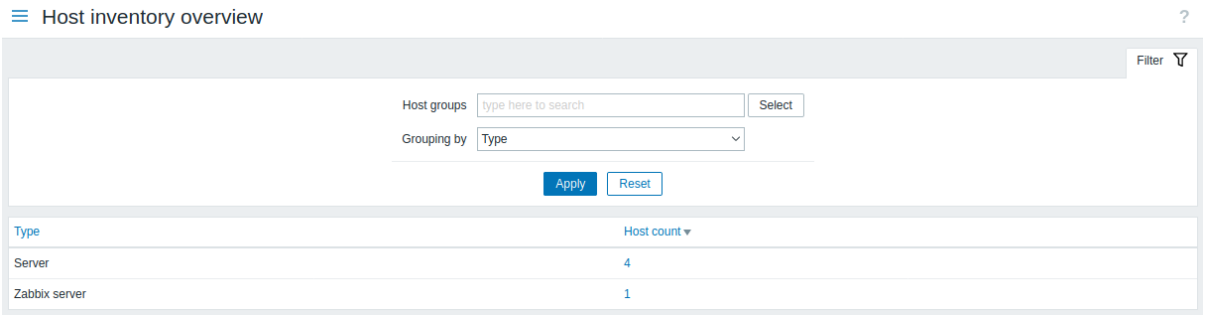
Menu Inwentaryzacja zawiera sekcje zapewniające przegląd danych inwentaryzacyjnych hostów według wybranego parametru, a także możliwość wyświetlania szczegółów inwentaryzacyjnych hosta.

Przegląd

Przegląd

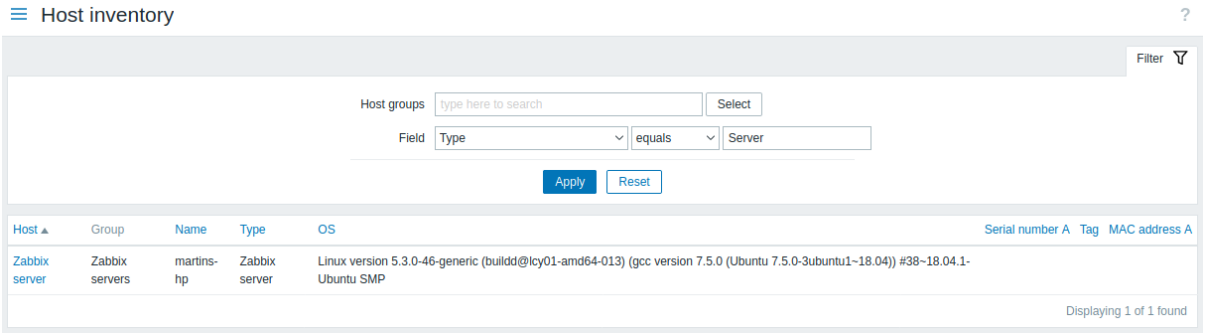
Sekcja *Inventory* > *Overview* zapewnia sposoby uzyskania przeglądu danych inwentarza hosta.

Aby wyświetlić przegląd, wybierz grupy hostów (lub żadną) oraz pole inwentarza, według którego mają być wyświetlane dane. Zostanie wyświetlona liczba hostów odpowiadających każdemu wpisowi wybranego pola.



Kompletność przeglądu zależy od tego, jak dużo informacji inwentarza jest utrzymywanych dla hostów.

Liczby w kolumnie *Host count* są linkami; prowadzą do hostów odfiltrowanych w tabeli *Host Inventories*.

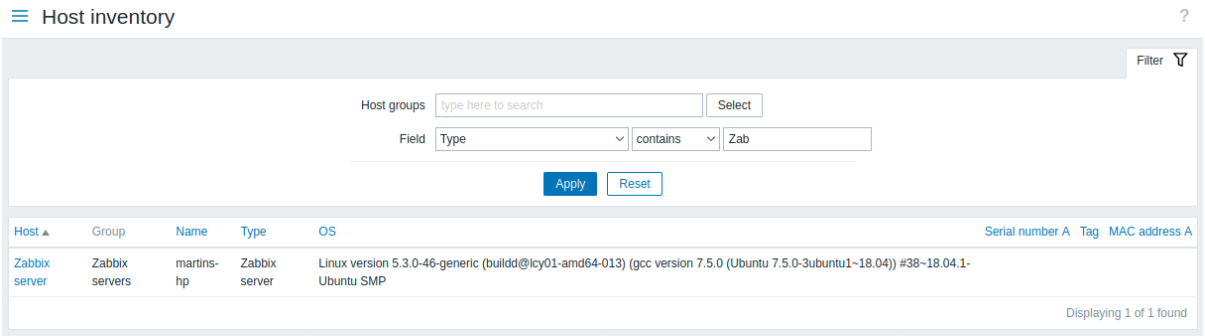


Hosty

Przegląd

W sekcji *Inventory* > *Hosts* dane inwentarzowe hostów są wyświetlane.

Możesz filtrować hosty według grup hostów i dowolnego pola inwentarzowego, aby wyświetlić tylko hosty, które Cię interesują.

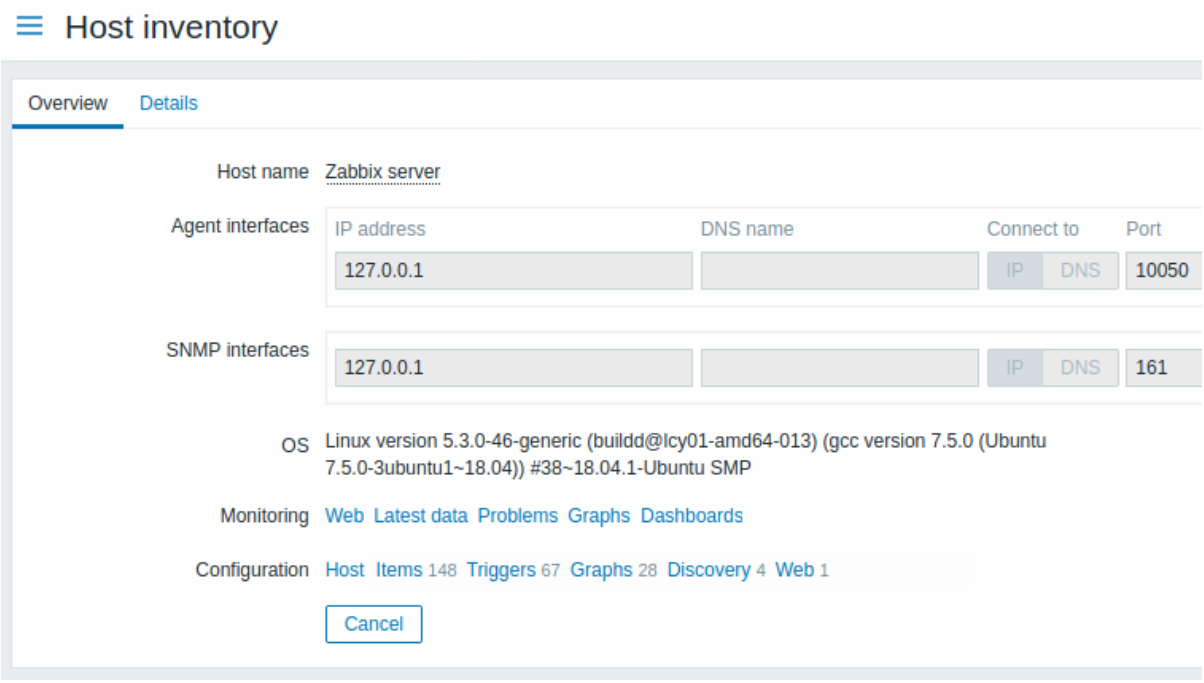


Aby wyświetlić wszystkie inwentarze hostów, kliknij przycisk *Reset*.

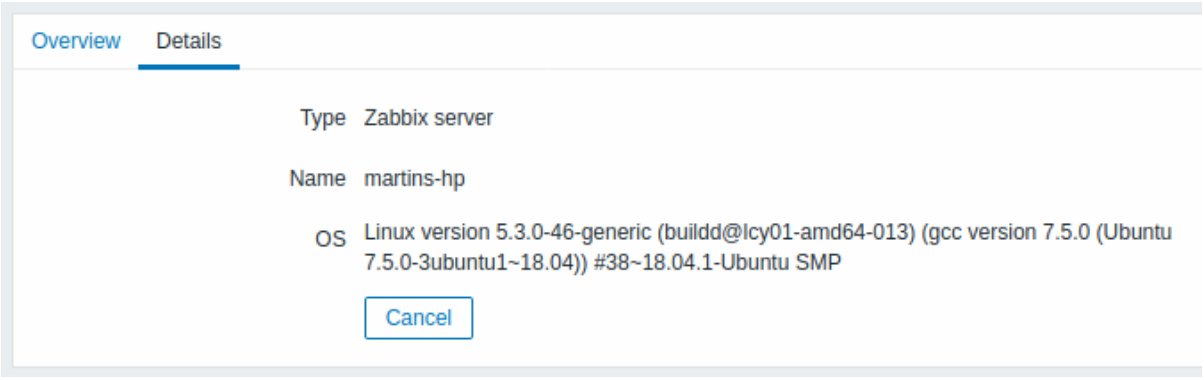
Chociaż w tabeli wyświetlane są tylko niektóre kluczowe pola inwentarzowe, możesz również zobaczyć wszystkie dostępne informacje inwentarzowe dla tego hosta. Aby to zrobić, kliknij nazwę hosta w pierwszej kolumnie.

Szczegóły inwentarza

Zakładka **Przegląd** zawiera ogólne informacje o hoście, najnowsze dane monitoringu oraz opcje konfiguracji hosta:



Zakładka **Szczegóły** zawiera wszystkie dostępne szczegóły inwentarza dla hosta:



Kompletność danych inwentarza zależy od tego, ile informacji inwentarzowych jest utrzymywanych dla hosta. Jeśli żadne informacje nie są utrzymywane, zakładka *Szczegóły* jest wyłączona.

Raporty

Przegląd

Menu Raporty zawiera kilka sekcji z różnorodnymi, predefiniowanymi i dostosowywanymi przez użytkownika raportami, które koncentrują się na wyświetlaniu przeglądu takich parametrów jak informacje o systemie, wyzwalacze i zebrane dane.

Informacje o systemie

Przegląd

W sekcji *Raporty* > *Informacje o systemie* wyświetlane jest podsumowanie najważniejszych danych dotyczących serwera Zabbix i systemu. Dane systemowe są zbierane za pomocą wewnętrznych pozycji.

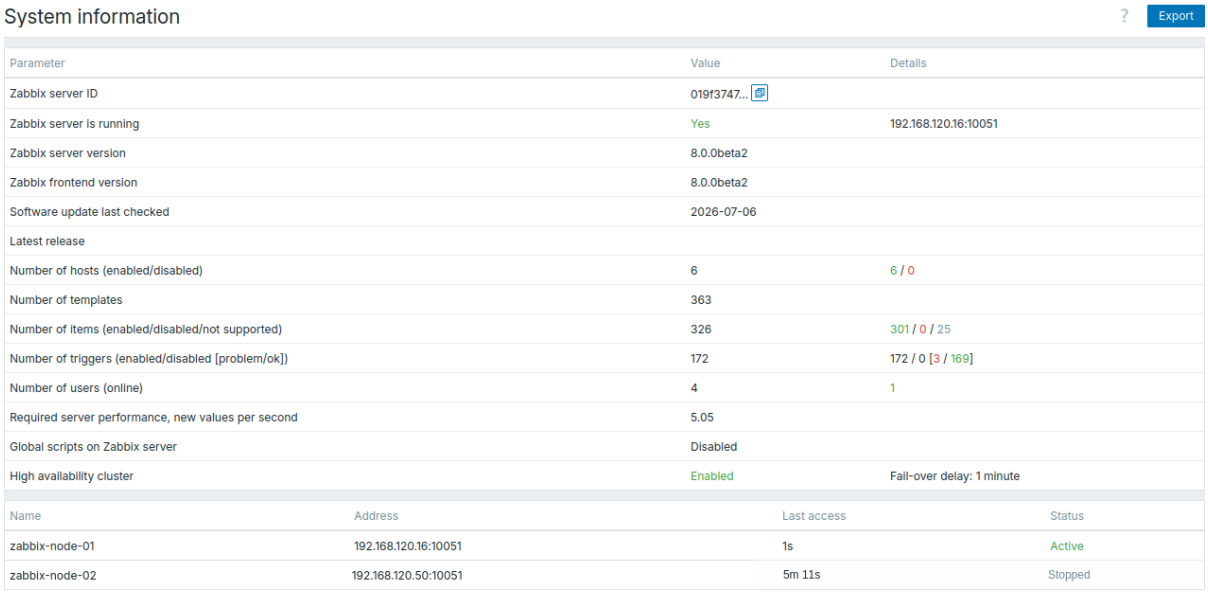
Raport z informacjami o systemie jest widoczny tylko dla użytkowników Zabbix z rolą *Super Admin* i *Admin*.

Należy pamiętać, że w konfiguracji wysokiej dostępności można przekierować źródło informacji o systemie (instancję serwera). W tym celu należy edytować plik konfiguracji frontendu - usunąć znak komentarza i ustawić `$ZBX_SERVER` lub zarówno `$ZBX_SERVER`, jak i `$ZBX_SERVER_PORT` na serwer inny niż aktualnie aktywny. Należy pamiętać, że w przypadku ustawienia tylko `$ZBX_SERVER` zostanie użyta wartość domyślna (10051) dla `$ZBX_SERVER_PORT`.

Po włączeniu konfiguracji wysokiej dostępności poniżej statystyk systemu wyświetlany jest osobny blok ze szczegółami dotyczącymi węzłów wysokiej dostępności.

Informacje o systemie są również dostępne jako widżet pulpitu nawigacyjnego.

Statystyki systemu



Wyświetlane dane:

Parametr	Wartość	Szczegóły
Identyfikator serwera Zabbix	Unikatowy identyfikator serwera Zabbix (UUIDv7).	Identyfikator serwera jest generowany przy pierwszym uruchomieniu serwera. Wyświetlanych jest pierwszych 8 znaków, po których następuje wielokropek. Po najechaniu kursorem na wartość identyfikatora serwera lub jej kliknięciu pojawia się etykieta narzędzia z pełną wartością. Kliknięcie ikony obok wartości kopiuje identyfikator do schowka. Jeśli włączona jest wysoka dostępność, każdy węzeł ma własny identyfikator węzła używany do adresowania z wiersza poleceń, ale wszystkie węzły są zgrupowane pod jednym identyfikatorem serwera wygenerowanym przez węzeł uruchomiony jako pierwszy (lub wcześniej działający jako samodzielny serwer), niezależnie od tego, który węzeł jest obecnie aktywny.
Serwer Zabbix jest uruchomiony	Stan serwera Zabbix: **Tak** - serwer jest uruchomiony **Nie** - serwer nie jest uruchomiony *Uwaga:* Aby wyświetlić pozostałe informacje, frontend webowy potrzebuje uruchomionego serwera oraz co najmniej jednego uruchomionego procesu trappera na serwerze (parametr StartTrappers w pliku `zabbix_server.conf` > 0).	Lokalizacja i port serwera Zabbix.
Wersja serwera Zabbix	Wyświetlany jest numer bieżącej wersji serwera. *Uwaga:* Jest wyświetlany tylko wtedy, gdy serwer Zabbix jest uruchomiony.	Wyświetlany jest stan wersji serwera: **Aktualna** - używana jest najnowsza wersja; **Dostępna nowa aktualizacja** - dostępna jest nowsza wersja; **Nieaktualna** - upłynął pełny okres wsparcia dla tej wersji. Informacja ta jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix. Nic nie jest wyświetlane, jeśli ostatnie sprawdzenie aktualizacji oprogramowania wykonano ponad tydzień temu lub brak jest danych o bieżącej wersji.

Parametr	Wartość	Szczegóły
Wersja frontendu Zabbix	Wyświetlany jest numer wersji frontendu Zabbix.	Wyświetlany jest stan wersji frontendu Zabbix: **Aktualna** - używana jest najnowsza wersja; **Dostępna nowa aktualizacja** - dostępna jest nowsza wersja; **Nieaktualna** - upłynął pełny okres wsparcia dla tej wersji. Informacja ta jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera Zabbix. Nic nie jest wyświetlane, jeśli ostatnie sprawdzenie aktualizacji oprogramowania wykonano ponad tydzień temu lub brak jest danych o bieżącej wersji.
Ostatnie sprawdzenie aktualizacji oprogramowania	Wyświetlana jest data ostatniego sprawdzenia aktualizacji oprogramowania Zabbix. Informacja ta jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera.	
Najnowsze wydanie	Wyświetlany jest numer nowszego wydania (jeśli jest dostępne) dla bieżącej wersji Zabbix. Informacja ta jest dostępna tylko wtedy, gdy sprawdzanie aktualizacji oprogramowania jest włączone w konfiguracji serwera. Nic nie jest wyświetlane, jeśli ostatnie sprawdzenie aktualizacji oprogramowania wykonano ponad tydzień temu lub brak jest danych o bieżącej wersji.	Wyświetlany jest link do informacji o wydaniu najnowszej dostępnej wersji Zabbix.
Liczba hostów	Wyświetlana jest łączna liczba skonfigurowanych hostów.	Liczba monitorowanych/niemonitorowanych hostów.
Liczba szablonów	Wyświetlana jest łączna liczba szablonów.	
Liczba pozycji	Wyświetlana jest łączna liczba pozycji.	Liczba monitorowanych/wyłączonych/nieobsługiwanych pozycji na poziomie hosta. Pozycje na wyłączonych hostach są liczone jako wyłączone.
Liczba wyzwalaczy	Wyświetlana jest łączna liczba wyzwalaczy.	Liczba włączonych/wyłączonych wyzwalaczy na poziomie hosta; podział włączonych wyzwalaczy według stanów „Problem”/„OK”. Wyzwalacze wymienione w stanie „OK” obejmują wyzwalacze ze stanem „Nieznany”. Wyzwalacze zależne od wyłączonych pozycji lub przypisane do wyłączonych hostów są liczone jako wyłączone.
Liczba użytkowników	Wyświetlana jest łączna liczba skonfigurowanych użytkowników.	Liczba użytkowników online.
Wymagana wydajność serwera, nowe wartości na sekundę	Wyświetlana jest oczekiwana liczba nowych wartości przetwarzanych przez serwer Zabbix na sekundę.	*Wymagana wydajność serwera* jest wartością szacunkową i może służyć jako wskazówka. Aby uzyskać dokładne liczby przetwarzanych wartości, użyj `zabbix[wcache,values,all]` wewnętrznej pozycji. W obliczeniach uwzględniane są włączone pozycje z monitorowanych hostów. Pozycje dziennika są liczone jako jedna wartość na każdy interwał aktualizacji pozycji. Uwzględniane są wartości z regularnych interwałów; wartości z elastycznych i harmonogramowanych interwałów nie są uwzględniane. Obliczenia nie są korygowane w okresie konserwacji „nodata”. Pozycje trappera nie są uwzględniane.

Parametr	Wartość	Szczegóły
Skrypty globalne na serwerze Zabbix	W tym polu wyświetlana jest wartość **Wyłączone**, jeśli skrypty globalne są wyłączone na serwerze Zabbix przez ustawienie EnableGlobalScripts=0 w konfiguracji serwera.	
Klaster wysokiej dostępności	Stan klastra wysokiej dostępności dla serwera Zabbix: **Wyłączony** - samodzielny serwer **Włączony** - istnieje co najmniej jeden węzeł wysokiej dostępności	Jeśli jest włączony, wyświetlane jest opóźnienie przełączenia awaryjnego.

Informacje o systemie wyświetlają również komunikat o błędzie w następujących przypadkach:

- Używana baza danych nie ma wymaganego zestawu znaków lub sortowania (UTF-8).
- Wersja bazy danych jest niższa lub wyższa niż obsługiwany zakres (dostępne tylko dla użytkowników z typem roli *Super admin*).
- Housekeeping dla TimescaleDB jest nieprawidłowo skonfigurowany (tabele historii lub trendów zawierają skompresowane fragmenty, ale opcje *Override item history period* lub *Override item trend period* są wyłączone).

Przyciski

Przycisk w prawym górnym rogu strony oferuje następującą opcję:

Export	Eksportuje informacje o systemie do pliku JSON. Plik jest zapisywany jako `Zabbix-system-information-{time}.json`, gdzie `{time}` to znacznik czasu Unix wskazujący, kiedy dane zostały przygotowane na serwerze. Jeśli włączona jest wysoka dostępność, eksportowane są również szczegóły węzłów wysokiej dostępności.

Węzły wysokiej dostępności

Jeśli klaster wysokiej dostępności jest włączony, wyświetlany jest dodatkowy blok danych pokazujący stan każdego węzła wysokiej dostępności.



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 2021-10-20 17:58:47	Standby

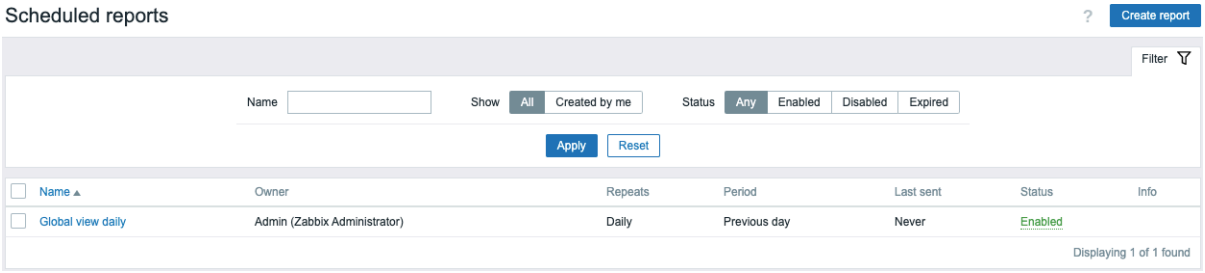
Wyświetlane dane:

Column	Description
Name	Nazwa węzła, zgodnie z konfiguracją serwera.
Address	Adres IP i port węzła.
Last access	Czas ostatniego dostępu do węzła. Po najechaniu kursorem na komórkę wyświetlany jest znacznik czasu ostatniego dostępu w długim formacie.
Status	Status węzła. Wiersze tabeli są sortowane według tych statusów w następującej kolejności priorytetowej: **Active** - węzeł działa i pracuje **Unavailable** - węzeł nie był widziany przez czas dłuższy niż opóźnienie przełączenia awaryjnego (warto ustalić przyczynę) **Stopped** - węzeł został zatrzymany lub nie mógł się uruchomić (warto go uruchomić lub usunąć) **Standby** - węzeł działa i oczekuje

Zaplanowane raporty

Przegląd

W sekcji *Raporty* > *Raporty zaplanowane* użytkownicy z odpowiednimi uprawnieniami mogą skonfigurować zaplanowane generowanie wersji PDF pulpitów nawigacyjnych, które będą wysyłane pocztą elektroniczną do określonych odbiorców.



Na ekranie początkowym wyświetlane są informacje o raportach zaplanowanych, które można filtrować w celu ułatwienia nawigacji — zobacz sekcję Korzystanie z filtra poniżej.

Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa raportu. Kliknięcie jej otwiera formularz konfiguracji raportu.
Właściciel	Użytkownik, który utworzył raport.
Powtarzanie	Częstotliwość generowania raportu (codziennie/co tydzień/co miesiąc/co rok).
Okres	Okres, za który przygotowywany jest raport (poprzedni dzień, tydzień, miesiąc lub rok).
Ostatnio wysłany	Data i godzina wysłania najnowszego raportu.
Status	Bieżący status raportu (włączony/wyłączony/wygasły). Użytkownicy z odpowiednimi uprawnieniami mogą zmienić status, klikając go — z „Włączony" na „Wyłączony" (i odwrotnie) oraz z „Wygasły" na „Wyłączony" (i odwrotnie). W przypadku użytkowników bez wystarczających uprawnień statusu nie można kliknąć.
Informacje	Wyświetla ikony informacyjne: Czerwona ikona oznacza, że generowanie raportu nie powiodło się; umieszczenie nad nią wskaźnika myszy wyświetli etykietę narzędzia z informacjami o błędzie. Żółta ikona oznacza, że raport został wygenerowany, ale wysłanie go do niektórych lub wszystkich odbiorców nie powiodło się albo raport wygasł; umieszczenie nad nią wskaźnika myszy wyświetli etykietę narzędzia z dodatkowymi informacjami.

Używanie filtra

Możesz użyć filtra, aby zawęzić listę raportów. Dla lepszej wydajności wyszukiwania dane są przeszukiwane przy nierozwiniętych makrach.

Dostępne są następujące opcje filtrowania:

- *Nazwa* - dozwolone jest częściowe dopasowanie nazwy
- *Pokaż* - utworzone przez bieżącego użytkownika lub wszystkie raporty
- *Status* - wybierz spośród „Dowolny" (pokaż wszystkie raporty), „Włączony", „Wyłączony" lub „Wygasły"

Filtr znajduje się poniżej nazwy sekcji *Raporty cykliczne*. Można go rozwinąć i zwinąć, klikając kartę *Filtr* w prawym górnym rogu.

Aktualizacja masowa

Czasami możesz chcieć usunąć lub zmienić status wielu raportów jednocześnie. Zamiast otwierać każdy raport osobno do edycji, możesz w tym celu użyć funkcji aktualizacji masowej.

Aby masowo zaktualizować raporty, wykonaj następujące czynności:

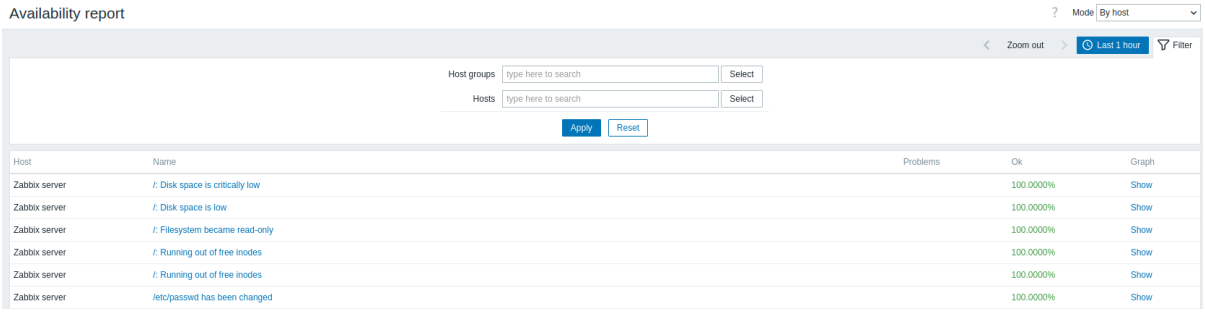
- Zaznacz pola wyboru raportów, które chcesz zaktualizować, na liście
- Kliknij odpowiedni przycisk pod listą, aby wprowadzić zmiany (*Włącz*, *Wyłącz* lub *Usuń*)

Raport dostępności

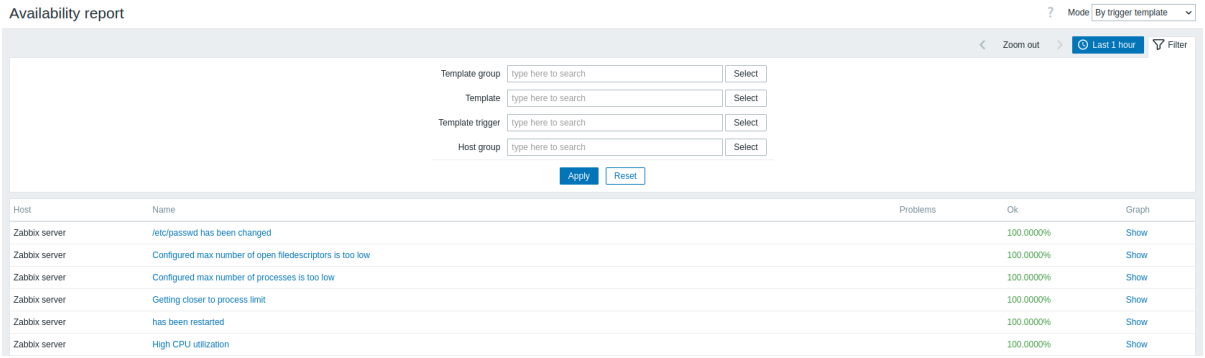
Przegląd

W *Raporty* > *Raport dostępności* możesz zobaczyć, przez jaki procent czasu każdy wyzwalacz znajdował się w stanie problem/OK.

Dla każdego stanu wyświetlany jest procent czasu, co ułatwia określenie dostępności różnych elementów w systemie.



Z listy rozwijanej w prawym górnym rogu możesz wybrać tryb wyboru — czy wyświetlać wyzwalacze według hostów, czy według wyzwalaczy należących do szablonu.



Nazwa wyzwalacza jest łączem do najnowszych zdarzeń tego wyzwalacza.

Używanie filtra

Filtr może pomóc zawęzić liczbę wyświetlanych hostów i/lub wyzwalaczy. Aby poprawić wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Filtr znajduje się poniżej nazwy sekcji *Availability report*. Można go otworzyć i zwinąć, klikając kartę *Filter* po prawej stronie.

Filtrowanie według szablonu wyzwalacza

W trybie *Według szablonu wyzwalacza* wyniki można filtrować według jednego lub kilku parametrów wymienionych poniżej.

Parametr	Opis
Grupa szablonów	Filtruj hosty według wyzwalaczy dziedziczonych z szablonów należących do wybranej grupy szablonów. Określenie nadrzędnej grupy szablonów powoduje niejawne wybranie wszystkich zagnieżdżonych grup szablonów.
Szablon	Filtruj hosty według wyzwalaczy dziedziczonych z wybranego szablonu, w tym z szablonów zagnieżdżonych. Jeśli zagnieżdżony szablon ma własne wyzwalacze, te wyzwalacze nie będą wyświetlane.
Wyzwalacz szablonu	Filtruj hosty według wybranego wyzwalacza. Inne wyzwalacze przefiltrowanych hostów nie będą wyświetlane.
Grupa hostów	Filtruj hosty należące do wybranej grupy hostów.

Filtrowanie według hosta

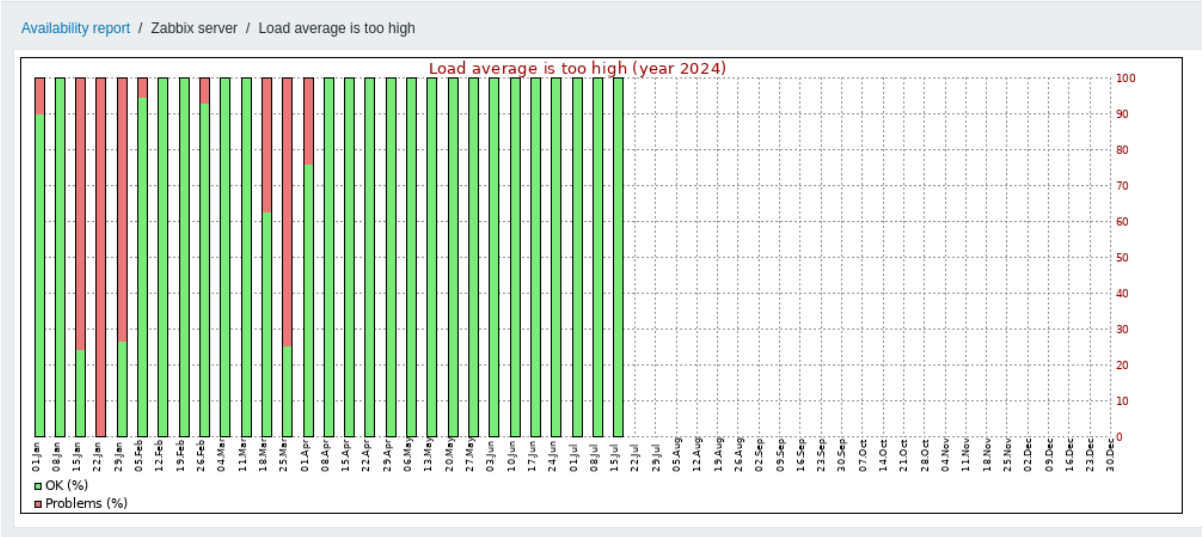
W trybie *Według hosta* wyniki mogą być filtrowane według hosta lub grupy hostów. Określenie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów.

Selektor okresu

Selektor okresu umożliwia wybranie często używanych okresów jednym kliknięciem. Selektor można rozwinąć i zwinąć, klikając kartę *Okres* znajdującą się obok filtra.

Kliknięcie opcji *Pokaż* w kolumnie *Wykres* wyświetla informacje o dostępności na wykresie słupkowym, na którym każdy słupek przedstawia miniony tydzień bieżącego roku.

Availability report graph



Kolor zielony słupka oznacza czas w stanie OK, a czerwony — czas problemu.

Wpływ okresów konserwacji

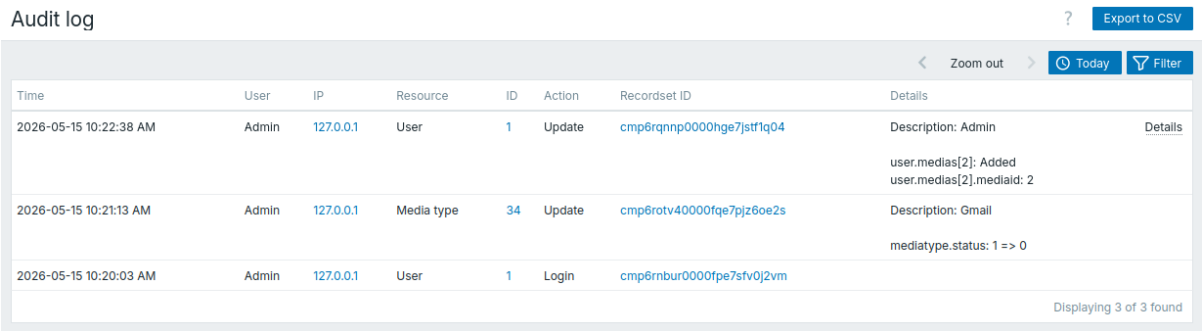
Konserwacja nie wyklucza automatycznie czasu z raportu dostępności. Tylko konserwacja skonfigurowana jako *No data collection* zatrzymuje odpytywanie (więc nie są generowane żadne problemy) i dlatego pozostawia procent dostępności bez zmian dla dotkniętych wyzwalaczy.

Dziennik audytu

Przegląd

W sekcji *Reports* > *Audit log* można przeglądać rekordy aktywności użytkowników i systemu.

Note:
Aby rekordy audytu były zbierane i wyświetlane, pole wyboru *Enable audit logging* musi być zaznaczone w sekcji *Administration* > *Audit log*. Bez włączenia tego ustawienia historia aktywności nie będzie zapisywana w bazie danych i nie będzie wyświetlana w dzienniku audytu.



Dziennik audytu wyświetla następujące dane:

Column	Description
Time	Znacznik czasu rekordu audytu.
User	Użytkownik, który wykonał aktywność.
IP	Adres IP, z którego zainicjowano aktywność. Kliknięcie hiperłącza spowoduje filtrowanie rekordów dziennika audytu według tego adresu IP.
Resource	Typ dotkniętego zasobu (*API token*, *Action*, *Authentication*, *Autoregistration*, itd.).
ID	ID dotkniętego zasobu. Kliknięcie hiperłącza spowoduje filtrowanie rekordów dziennika audytu według tego ID zasobu.
Action	Typ aktywności (*Add*, *Configuration refresh*, *Delete*, *Execute*, *Failed login*, *History clear*, *Login*, *Logout*, *Push*, *Update*).

Column	Description
Recordset ID	Wspólny ID dla wszystkich rekordów dziennika audytu utworzonych w wyniku tej samej operacji. Na przykład podczas powiązania szablonu z hostem dla każdej odziedziczonej encji szablonu (pozycja, wyzwalacz itd.) tworzony jest osobny rekord dziennika audytu - wszystkie te rekordy będą miały ten sam *Recordset ID*. Kliknięcie hiperłącza spowoduje filtrowanie rekordów dziennika audytu według tego *Recordset ID*.
Details	Opis zasobu i szczegółowe informacje o wykonanej aktywności. Jeśli rekord zawiera więcej niż dwa wiersze, zostanie wyświetlony dodatkowy link *Details*. Kliknij ten link, aby wyświetlić pełną listę zmian.

Note:
Gdy pozycja trapper lub pozycja HTTP agent (z włączonym trappingiem) otrzyma jakieś dane, wpis w dzienniku audytu zostanie dodany tylko wtedy, gdy dane zostały wysłane metodą API `history.push`, a nie za pomocą narzędzia Zabbix sender.

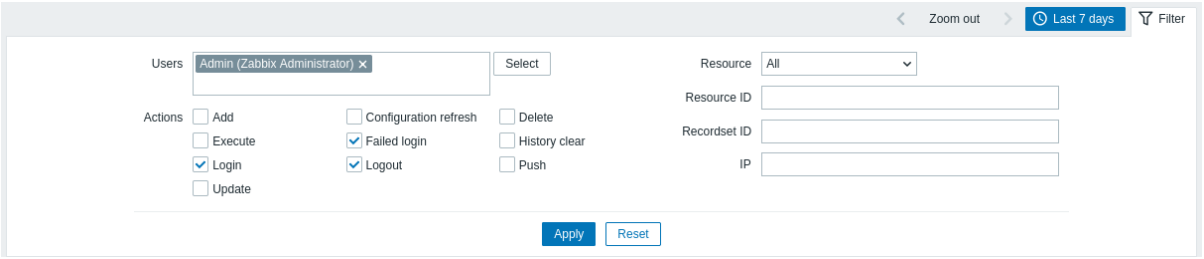
Przyciski

Przycisk w prawym górnym rogu strony oferuje następującą opcję:

Export to CSV	Eksportuj rekordy dziennika audytu ze wszystkich stron do pliku CSV. Jeśli zastosowano filtr, zostaną wyeksportowane tylko przefiltrowane rekordy.

Korzystanie z filtra

Filtr znajduje się poniżej paska *Dziennik audytu*. Można go otworzyć i zwinąć, klikając kartę *Filtr* w prawym górnym rogu.



Za pomocą filtra można zawęzić rekordy według użytkownika, zasobu, którego dotyczy operacja, identyfikatora zasobu, wykonanej operacji (*ID zestawu rekordów*) oraz adresu IP. W zależności od zasobu w filtrze można wybrać jedną lub więcej określonych akcji.

Aby zapewnić lepszą wydajność wyszukiwania, wszystkie dane są przeszukiwane z nierozwiniętymi makrami.

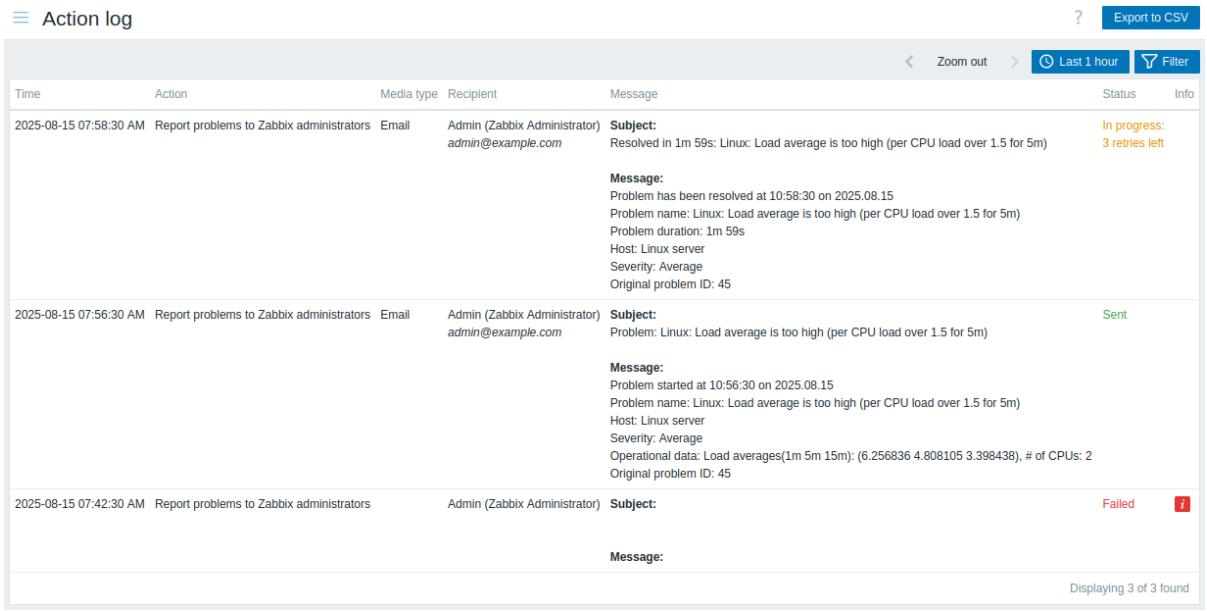
Selektor okresu

Selektor okresu obok filtra umożliwia wybranie często używanych okresów jednym kliknięciem myszy.
Więcej informacji można znaleźć w sekcji Selektory okresu i hosta.

Dziennik akcji

Przegląd

W *Reports* > *Action log* możesz wyświetlić szczegóły operacji (powiadomienia, zdalne polecenia) wykonanych w ramach akcji.



Wyświetlane dane:

Column	Description
Time	Znacznik czasu operacji.
Action	Nazwa akcji powodującej operacje.
Media type	Typ mediów (np. Email, Jira itp.) użyty do wysłania powiadomienia. W przypadku operacji, które wykonały zdalne polecenia, ta kolumna będzie pusta.
Recipient	Informacje o odbiorcy powiadomienia - nazwa użytkownika, imię i nazwisko (w nawiasach) oraz dodatkowe informacje zależnie od typu mediów (email, nazwa użytkownika itp.). W przypadku operacji, które wykonały zdalne polecenia, ta kolumna będzie pusta.
Message	Treść wiadomości, zdalnego polecenia lub nazwa skryptu globalnego. Zdalne polecenie jest oddzielone od hosta docelowego symbolem dwukropka: `<host>:<command>`. Na przykład, jeśli zdalne polecenie zostało wykonane na serwerze Zabbix, informacja będzie miała następujący format: `Zabbix server:<command>`.
Status	Status operacji: *In progress* - operacja wysyłania powiadomienia jest w toku (wyświetlana jest również pozostała liczba prób, jakie serwer podejmie, aby wysłać powiadomienie) *Sent* - powiadomienie zostało wysłane *Executed* - zdalne polecenie zostało wykonane *Failed* - operacja nie została zakończona
Info	Informacje o błędzie (jeśli występują) dotyczące wykonania operacji.

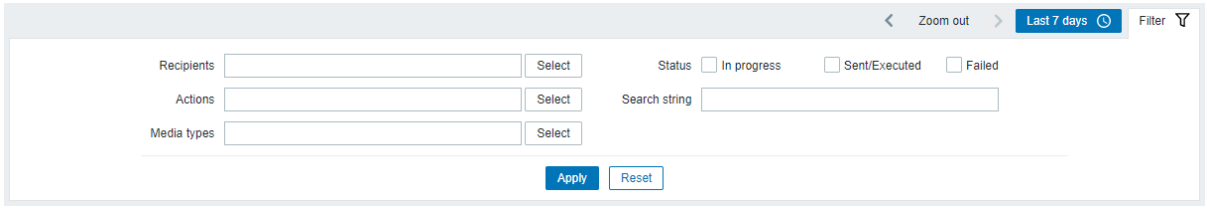
Przyciski

Przycisk w prawym górnym rogu strony oferuje następującą opcję:

Export to CSV	Eksportuj rekordy dziennika akcji z bieżącej strony do pliku CSV. Jeśli zastosowano filtr, wyeksportowane zostaną tylko przefiltrowane rekordy. W wyeksportowanym pliku CSV kolumny „Recipient" i „Message" są podzielone na kilka kolumn — „Recipient's Zabbix username", „Recipient's name", „Recipient's surname", „Recipient" oraz „Subject", „Message", „Command".

Korzystanie z filtra

Filtr znajduje się poniżej paska *Action log*. Można go otworzyć i zwinąć, klikając kartę *Filter* w prawym górnym rogu strony.



Możesz użyć filtra, aby zawęzić rekordy według odbiorców powiadomień, akcji, typów mediów, statusu lub według zawartości wiadomości/polecenia zdalnego (*Search string*). Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Selektor okresu

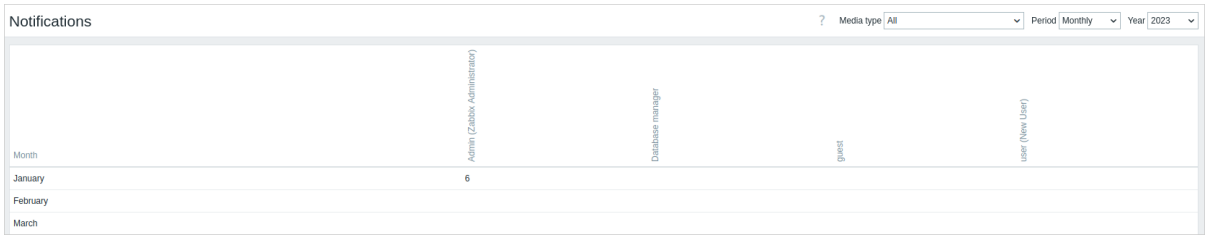
Selektor okresu obok filtra umożliwia wybranie często używanych okresów jednym kliknięciem myszy.
Więcej informacji można znaleźć w sekcji Selektory okresu i hosta.

Powiadomienia

Przegląd

W sekcji *Reports* > *Notifications* wyświetlany jest raport dotyczący liczby powiadomień wysłanych do każdego użytkownika.

Z list rozwijanych w prawym górnym rogu można wybrać typ mediów (lub wszystkie), okres (dane dla każdego dnia/tygodnia/miesiąca/roku) oraz rok dla wysłanych powiadomień.

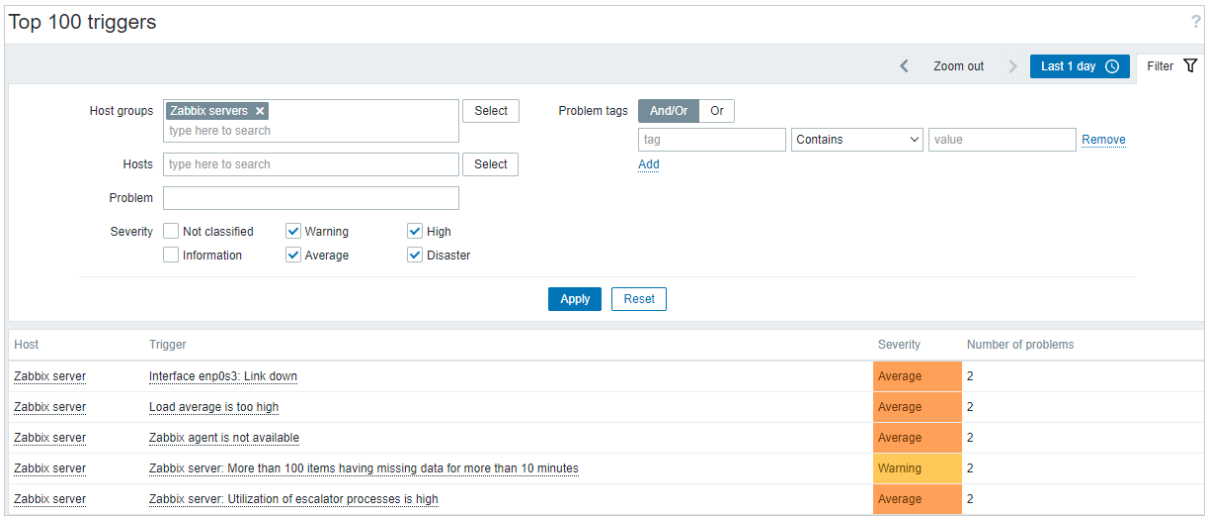


Każda kolumna wyświetla sumy dla jednego użytkownika systemu.

100 najważniejszych wyzwalaczy

Przegląd

W sekcji *Raporty* > *100 najczęstszych wyzwalaczy* można wyświetlić wyzwalacze z największą liczbą problemów wykrytych w wybranym okresie.



Wpisy w kolumnach hosta i wyzwalacza są linkami oferującymi przydatne opcje:

- dla hosta - kliknięcie nazwy hosta otwiera menu hosta
- dla wyzwalacza - kliknięcie nazwy wyzwalacza otwiera linki do najnowszych zdarzeń, prostego wykresu dla każdej pozycji wyzwalacza oraz formularzy konfiguracji samego wyzwalacza i każdej pozycji wyzwalacza

Korzystanie z filtra

Filtr umożliwia wyświetlanie wyzwalaczy według grupy hostów, hosta, nazwy problemu, znaczników lub ważności wyzwalacza. Określenie nadrzędnej grupy hostów powoduje niejawne wybranie wszystkich zagnieżdżonych grup hostów. Aby zwiększyć wydajność wyszukiwania, dane są wyszukiwane z nierozwiniętymi makrami.

Filtr znajduje się poniżej nazwy sekcji *100 najczęstszych wyzwalaczy*. Można go otworzyć i zwinąć, klikając kartę *Filtr* po prawej stronie.

Selektor okresu

Selektor okresu obok filtra umożliwia wybranie często używanych okresów jednym kliknięciem myszy. Więcej informacji można znaleźć w sekcji Selektory okresu i hosta.

Zbieranie danych

Przegląd

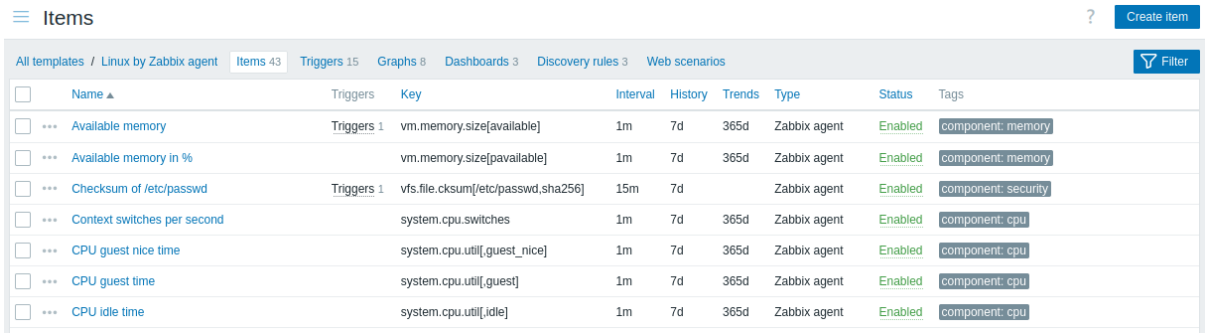
To menu zawiera sekcje związane z konfiguracją zbierania danych.

Pozycje

Przegląd

Lista pozycji dla szablonu jest dostępna w sekcji *Zbieranie danych* > *Szablony* po kliknięciu opcji *Pozycje* przy odpowiednim szablonie.

Wyświetlana jest lista istniejących pozycji.



Wyświetlane dane:

Kolumna	Opis
Menu pozycji	Kliknij ikonę z trzema kropkami, aby otworzyć menu pozycji.
Szablon	Szablon, do którego należy pozycja. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele szablonów lub nie wybrano żadnego szablonu.
Nazwa	Nazwa pozycji wyświetlana jako niebieski odnośnik do szczegółów pozycji. Kliknięcie odnośnika z nazwą pozycji otwiera formularz konfiguracji pozycji. Jeśli pozycja jest dziedziczona z innego szablonu, przed nazwą pozycji wyświetlana jest nazwa szablonu jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę pozycji na poziomie tego szablonu.
Wyzwalacze	Umieszczenie wskaźnika myszy nad opcją Wyzwalacze wyświetla pole informacyjne z wyzwalaczami powiązanymi z pozycją. Liczba wyzwalaczy jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania.
Historia	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trendy	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Typ	Wyświetlany jest typ pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itd.).
Stan	Wyświetlany jest stan pozycji — *Włączona* lub *Wyłączona*. Klikając stan, można go zmienić z Włączona na Wyłączona (i odwrotnie).
Znaczniki	Wyświetlane są znaczniki pozycji. Można wyświetlić maksymalnie trzy znaczniki (pary nazwa:wartość). Jeśli znaczników jest więcej, wyświetlany jest odnośnik „...", który umożliwia wyświetlenie wszystkich znaczników po umieszczeniu nad nim wskaźnika myszy.

Aby skonfigurować nową pozycję, kliknij przycisk *Utwórz pozycję* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

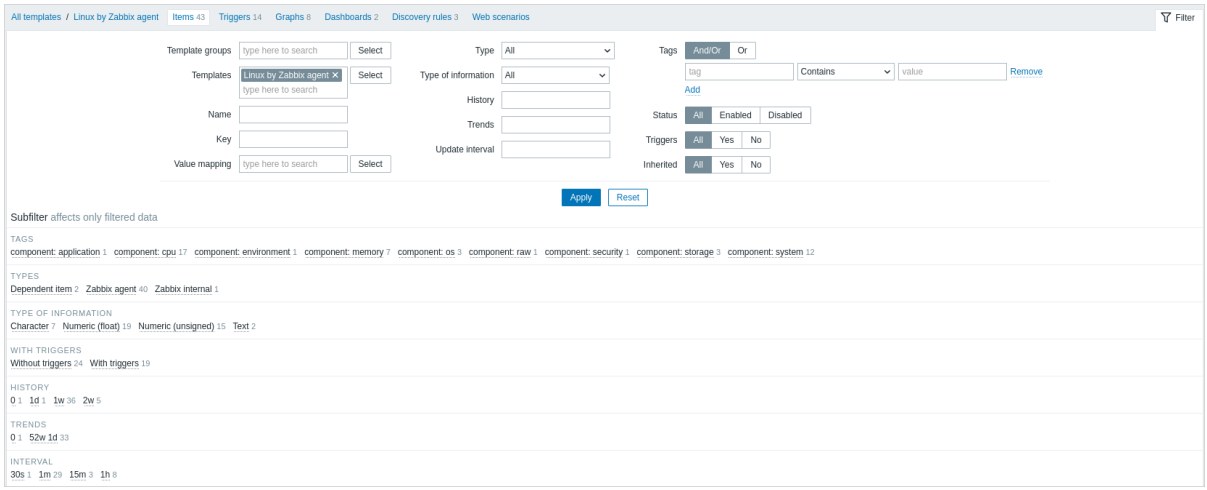
- *Enable* - zmień status pozycji na *Enabled*.
- *Disable* - zmień status pozycji na *Disabled*.
- *Copy* - skopiuj pozycje do innych hostów lub szablonów.
- *Mass update* - zaktualizuj kilka właściwości dla wielu pozycji jednocześnie.
- *Delete* - usuń pozycje.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi pozycjami, a następnie kliknij wymagany przycisk.

Używanie filtra

Lista pozycji może zawierać wiele elementów. Korzystając z filtra, możesz odfiltrować część z nich, aby szybko znaleźć pozycje, których szukasz. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym możesz określić żądane kryteria filtrowania.

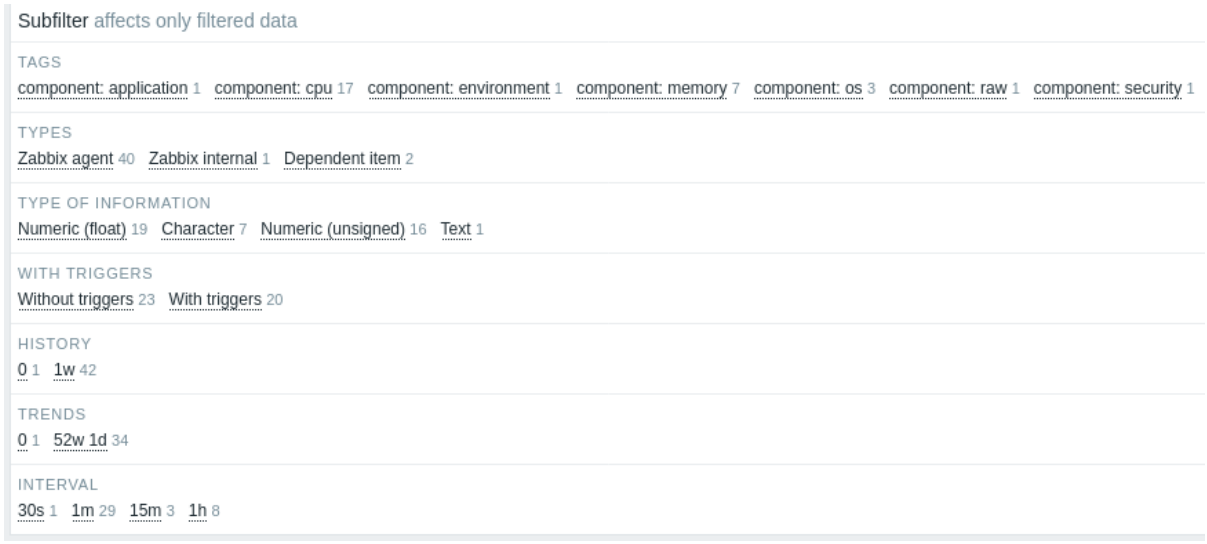


Parameter	Description
Template groups	Filtrowanie według jednej lub wielu grup szablonów. Określenie nadrzędnej grupy szablonów domyślnie zaznacza wszystkie zagnieżdżone grupy.
Templates	Filtrowanie według jednego lub wielu szablonów.
Name	Filtrowanie według nazwy pozycji.
Key	Filtrowanie według klucza pozycji.
Value mapping	Filtrowanie według używanego mapowania wartości. Ten parametr nie jest wyświetlany, jeśli opcja *Templates* jest pusta.
Type	Filtrowanie według typu pozycji (Zabbix agent, SNMP agent itp.).
Type of information	Filtrowanie według typu informacji (Numeric unsigned, float itp.).
History	Filtrowanie według czasu przechowywania historii pozycji.
Trends	Filtrowanie według czasu przechowywania trendów pozycji.
Update interval	Filtrowanie według interwału aktualizacji pozycji.
Tags	Określ tagi, aby ograniczyć liczbę wyświetlanych pozycji. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczeń dla warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek
Status	Filtrowanie według statusu pozycji - *Enabled* lub *Disabled*.
Triggers	Filtrowanie pozycji z wyzwalaczami lub bez wyzwalaczy.
Inherited	Filtrowanie pozycji dziedziczonych lub niedziedziczonych z połączonych szablonów.

Korzystanie z podfiltra

Podfiltr umożliwia dalszą modyfikację filtrowania zastosowanego przez filtr główny.

Zawiera klikalne odnośniki zapewniające szybki dostęp do powiązanych elementów. Elementy są powiązane wspólną encją, taką jak znacznik, typ pozycji, interwał aktualizacji pozycji itp. Po kliknięciu encji zostaje ona wyróżniona szarym tłem, a pozycje są natychmiast filtrowane (nie trzeba klikać przycisku *Zastosuj* w filtrze głównym). Kliknięcie innej encji dodaje ją do przefiltrowanych wyników. Ponowne kliknięcie encji usuwa filtrowanie.



Podfiltry są generowane na podstawie przefiltrowanych danych, które są ograniczone do 1000 rekordów. Aby wyświetlić więcej rekordów w podfiltrze, należy zwiększyć wartość parametru *Limit for search and filter results* (w sekcji *Administration* > *General* > *GUI*).

W przeciwieństwie do filtra głównego podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zawierać aktualne informacje o dostępnych opcjach filtrowania i ich licznikach.

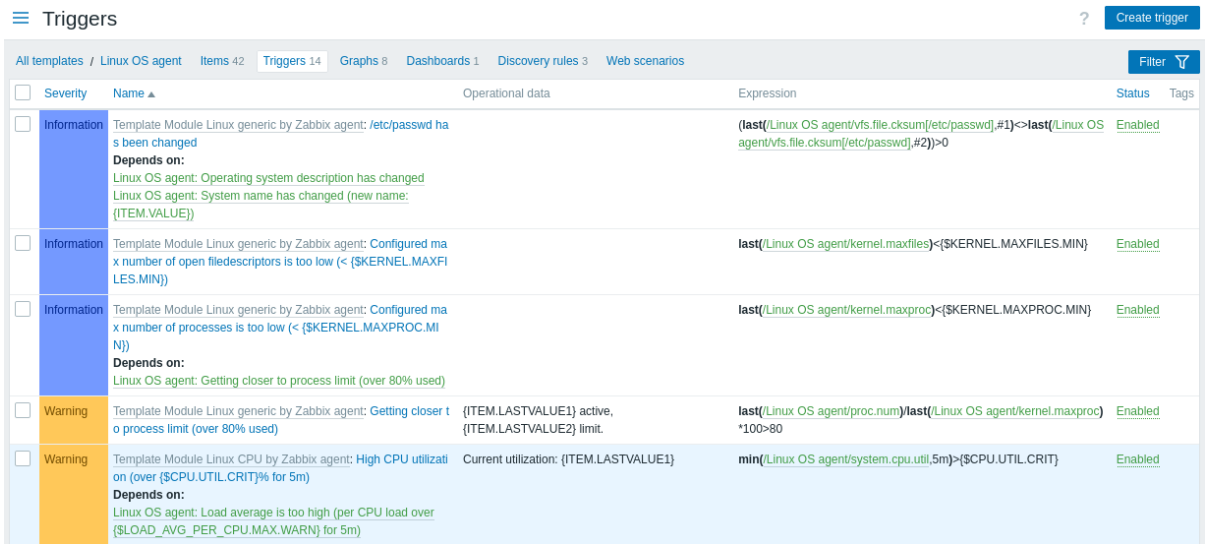
Liczba wyświetlanych encji jest ograniczona do 100 w poziomie. Jeśli jest ich więcej, na końcu wyświetlana jest ikona trzech kropek; nie można jej kliknąć.

Liczba obok każdej klikalnej encji wskazuje liczbę pogrupowanych w niej pozycji (na podstawie wyników filtra głównego). Po kliknięciu encji liczby przy innych dostępnych encjach są wyświetlane ze znakiem plus, wskazując, ile pozycji można dodać do bieżącego wyboru. Encje bez pozycji nie są wyświetlane, chyba że zostały wcześniej wybrane w podfiltrze.

Wyzwalacze

Przegląd

Lista wyzwalaczy dla szablonu jest dostępna w sekcji *Zbieranie danych* > *Szablony* po kliknięciu opcji *Wyzwalacze* przy odpowiednim szablonie.



Wyświetlane dane:

Kolumna	Opis
Poziom ważności	Poziom ważności wyzwalacza jest wyświetlany zarówno za pomocą nazwy, jak i koloru tła komórki.
Szablon	Szablon, do którego należy wyzwalacz. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele szablonów lub nie wybrano żadnego szablonu.
Nazwa	Nazwa wyzwalacza wyświetlana jako niebieski odnośnik do szczegółów wyzwalacza. Kliknięcie odnośnika z nazwą wyzwalacza otwiera formularz konfiguracji wyzwalacza. Jeśli wyzwalacz jest dziedziczony z innego szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary odnośnik. Kliknięcie odnośnika do szablonu otwiera listę wyzwalaczy na poziomie tego szablonu.
Dane operacyjne	Definicja danych operacyjnych wyzwalacza zawierająca dowolne ciągi znaków i makra, które zostaną dynamicznie rozwinięte w sekcji *Monitorowanie* > *Problemy*.
Wyrażenie	Wyświetlane jest wyrażenie wyzwalacza. Część wyrażenia dotycząca pozycji szablonu jest wyświetlana jako odnośnik prowadzący do formularza konfiguracji pozycji.
Stan	Wyświetlany jest stan wyzwalacza: *Włączony* lub *Wyłączony*. Klikając stan, można go zmienić z Włączony na Wyłączony i odwrotnie.
Znaczniki	Jeśli wyzwalacz zawiera znaczniki, w tej kolumnie wyświetlana jest nazwa znacznika i jego wartość.

Aby skonfigurować nowy wyzwalacz, kliknij przycisk *Utwórz wyzwalacz* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Enable* - zmiana statusu wyzwalacza na *Enabled*
- *Disable* - zmiana statusu wyzwalacza na *Disabled*
- *Copy* - skopiowanie wyzwalaczy do innych hostów lub szablonów
- *Mass update* - aktualizacja kilku właściwości dla wielu wyzwalaczy jednocześnie
- *Delete* - usunięcie wyzwalaczy

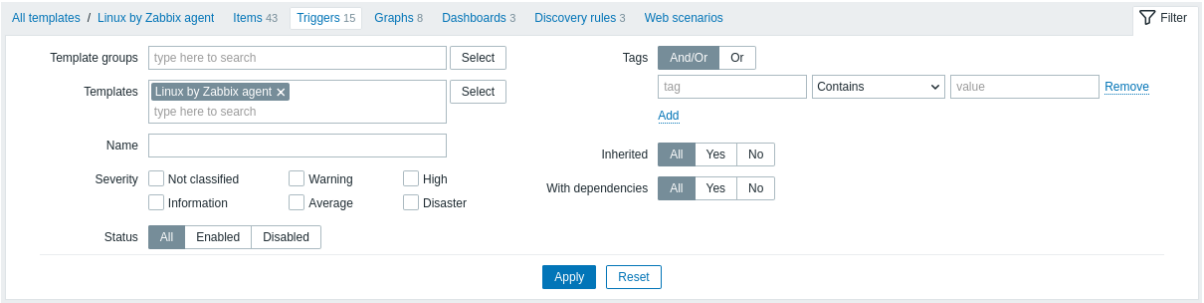
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi wyzwalaczami, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te wyzwalacze, które Cię interesują.

Dla lepszej wydajności wyszukiwania dane są przeszukiwane przy nierozwiniętych makrach.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym możesz określić żądane kryteria filtrowania.



Parameter	Description
Template groups	Filtruj według jednej lub kilku grup szablonów. Określenie nadrzędnej grupy szablonów automatycznie wybiera wszystkie zagnieżdżone grupy.
Templates	Filtruj według jednego lub kilku szablonów. Jeśli powyżej wybrano już grupy szablonów, wybór szablonów jest ograniczony do tych grup.
Name	Filtruj według nazwy wyzwalacza.
Severity	Wybierz, aby filtrować według jednej lub kilku wartości ważności wyzwalacza.
Status	Filtruj według statusu aktywacji wyzwalacza (Enabled/Disabled).

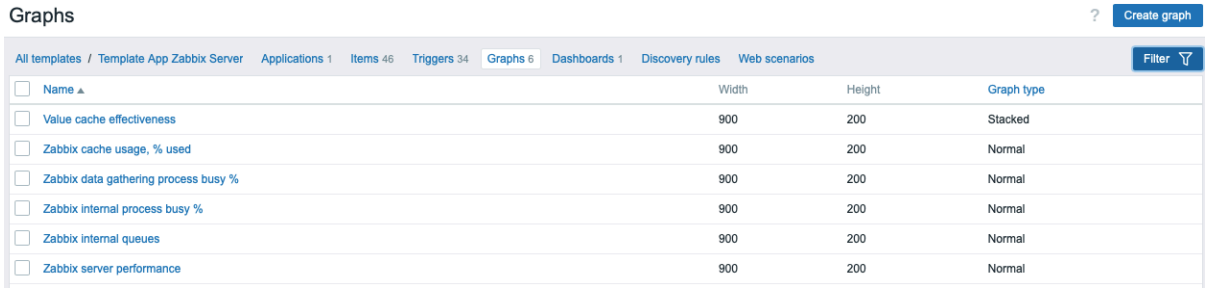
Parameter	Description
Tags	Filtruj według nazwy i wartości tagu wyzwalacza. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter) Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony jest jeden warunek Makra i funkcje makr są obsługiwane w polach nazwy tagu i wartości tagu.
Inherited	Filtruj wyzwalacze dziedziczone (lub niedziedziczone) z powiązanych szablonów.
With dependencies	Filtruj wyzwalacze z zależnościami (lub bez zależności).

Wykresy

Przegląd

Listę niestandardowych wykresów dla szablonu można otworzyć, wybierając *Gromadzenie danych* > *Szablony*, a następnie klikając *Wykresy* przy odpowiednim szablonie.

Wyświetlana jest lista istniejących wykresów.



Wyświetlane dane:

Kolumna	Opis
Szablon	Szablon, do którego należy wykres. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele szablonów lub nie wybrano żadnego szablonu.
Nazwa	Nazwa niestandardowego wykresu wyświetlana jako niebieski odnośnik do szczegółów wykresu. Kliknięcie odnośnika z nazwą wykresu otwiera formularz konfiguracji wykresu. Jeśli wykres jest dziedziczony z innego szablonu, przed nazwą wykresu wyświetlana jest nazwa szablonu jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę wykresów na poziomie tego szablonu.
Szerokość	Wyświetlana jest szerokość wykresu.
Wysokość	Wyświetlana jest wysokość wykresu.
Typ wykresu	Wyświetlany jest typ wykresu: *Normalny*, *Skumulowany*, *Kołowy* lub *Rozstrzelony*.

Aby skonfigurować nowy wykres, kliknij przycisk *Utwórz wykres* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Copy* - skopiuj wykresy do innych hostów lub szablonów
- *Delete* - usuń wykresy

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi wykresami, a następnie kliknij wymagany przycisk.

Używanie filtra

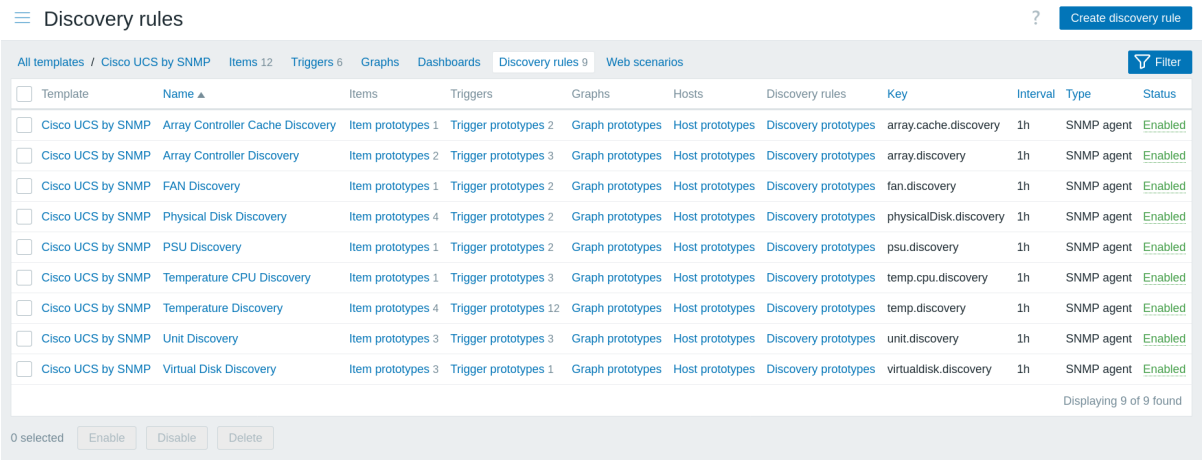
Możesz filtrować wykresy według grupy szablonów i szablonu. Aby uzyskać lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Reguły wykrywania

Przegląd

Lista reguł niskopoziomowego wykrywania dla szablonu jest dostępna w sekcji *Zbieranie danych* > *Szablony* po kliknięciu opcji *Wykrywanie* dla odpowiedniego szablonu.

Wyświetlana jest lista istniejących reguł niskopoziomowego wykrywania. Można również wyświetlić wszystkie reguły wykrywania niezależnie od szablonu lub wszystkie reguły wykrywania należące do określonej grupy szablonów, zmieniając ustawienia filtra.



Wyświetlane dane:

Kolumna	Opis
Szablon	Szablon, do którego należy reguła wykrywania. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu.
Nazwa	Nazwa reguły wyświetlana jako niebieski odnośnik. Kliknięcie nazwy reguły otwiera formularz konfiguracji reguły niskopoziomowego wykrywania. Jeśli reguła wykrywania jest dziedziczona z innego szablonu, nazwa szablonu jest wyświetlana przed nazwą reguły jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę reguł wykrywania na poziomie tego szablonu.
Pozycje	Wyświetlany jest odnośnik do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Wyzwalacze	Wyświetlany jest odnośnik do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Wykresy	Wyświetlany jest odnośnik do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosty	Wyświetlany jest odnośnik do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji używany do wykrywania.
Interwał	Wyświetlana jest częstotliwość wykonywania wykrywania.
Typ	Wyświetlany jest typ pozycji używany do wykrywania (agent Zabbix, agent SNMP itd.).
Stan	Wyświetlany jest stan reguły wykrywania — *Włączona* lub *Wyłączona*. Klikając stan, można go zmienić z Włączona na Wyłączona (i odwrotnie).

Aby skonfigurować nową regułę niskopoziomowego wykrywania, kliknij przycisk *Utwórz regułę wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

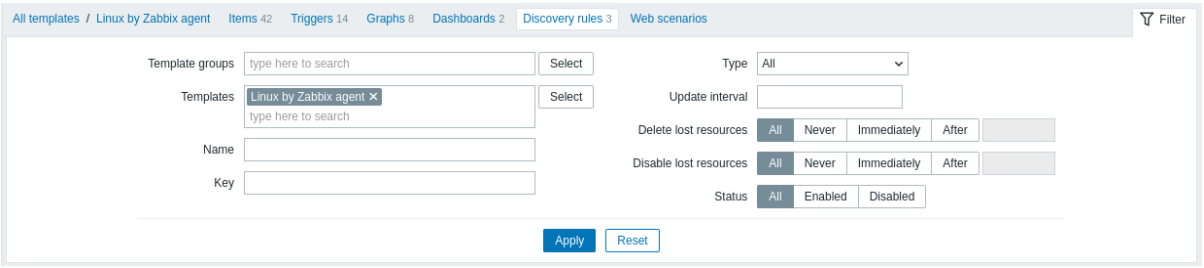
- *Włącz* - zmienia status reguły niskopoziomowego wykrywania na *Włączona*
- *Wyłącz* - zmienia status reguły niskopoziomowego wykrywania na *Wyłączona*
- *Usuń* - usuwa reguły niskopoziomowego wykrywania

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi regułami wykrywania, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły odkrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane przy nierozwiniętych makrach.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym można określić żądane kryteria filtrowania, takie jak szablon, nazwa reguły odkrywania, klucz pozycji, typ pozycji itp.



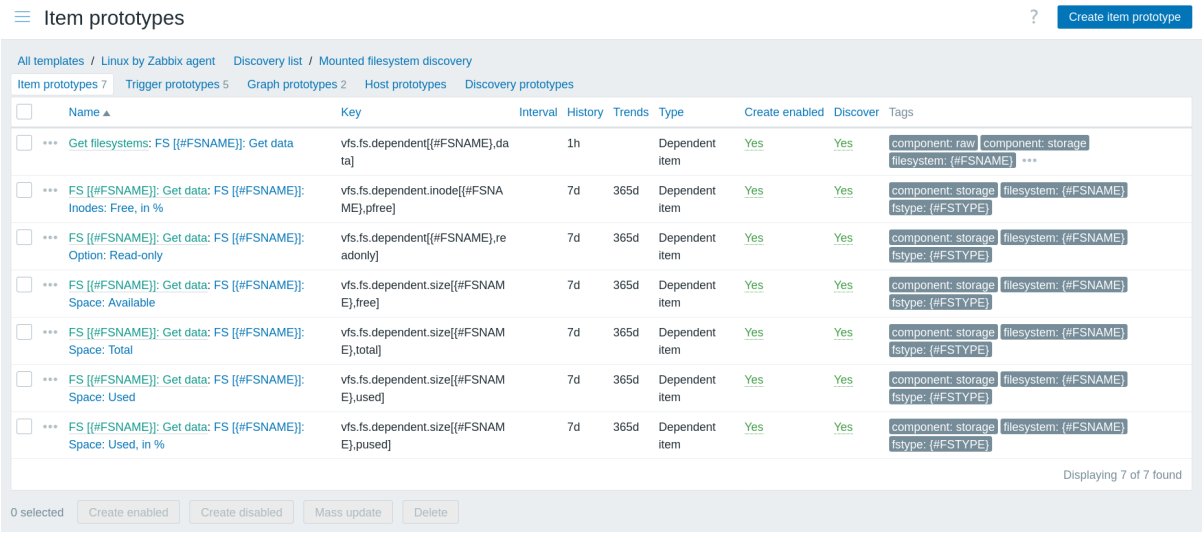
Parameter	Description
Template groups	Filtruj według jednej lub kilku grup szablonów. Określenie nadrzędnej grupy szablonów domyślnie zaznacza wszystkie zagnieżdżone grupy.
Templates	Filtruj według jednego lub kilku szablonów.
Name	Filtruj według nazwy reguły odkrywania.
Key	Filtruj według klucza pozycji odkrywania.
Type	Filtruj według typu pozycji odkrywania.
Update interval	Filtruj według interwału aktualizacji. Niedostępne dla Zabbix trapper i pozycji zależnych.
Delete lost resources	Filtruj według okresu *Delete lost resources*.
Disable lost resources	Filtruj według okresu *Disable lost resources*.
Status	Filtruj według statusu reguły odkrywania (All/Enabled/Disabled).

Prototypy pozycji

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy pozycji reguły wykrywania niskopoziomowego na szablonie.

Jeśli szablon jest powiązany z hostem, prototypy pozycji będą podstawą do tworzenia rzeczywistych pozycji hosta podczas wykrywania niskopoziomowego.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu pozycji wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu pozycji. Jeśli prototyp pozycji należy do powiązanego szablonu, nazwa szablonu jest wyświetlana przed nazwą prototypu pozycji jako szary link. Kliknięcie linku szablonu otwiera listę prototypów pozycji na poziomie powiązanego szablonu. Jeśli prototyp pozycji jest typu „Zależna pozycja”, nazwa pozycji nadrzędnej jest wyświetlana przed nazwą prototypu pozycji na zielono (tak jak w przypadku prototypów pozycji na poziomie hosta).
Klucz	Wyświetlany jest klucz prototypu pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania.
Historia	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trendy	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Typ	Wyświetlany jest typ prototypu pozycji (Zabbix agent, SNMP agent, proste sprawdzenie itd.).
Tworzenie włączone	Tworzenie pozycji na podstawie tego prototypu: **Tak** - włączone **Nie** - wyłączone. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Wykrywaj	Wykrywanie pozycji na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Znaczniki	Wyświetlane są znaczniki prototypu pozycji.

Aby skonfigurować nowy prototyp pozycji, kliknij przycisk *Utwórz prototyp pozycji* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz jako włączone* - utwórz te pozycje jako *Włączone*
- *Utwórz jako wyłączone* - utwórz te pozycje jako *Wyłączone*
- *Masowa aktualizacja* - masowo zaktualizuj te prototypy pozycji
- *Usuń* - usuń te prototypy pozycji

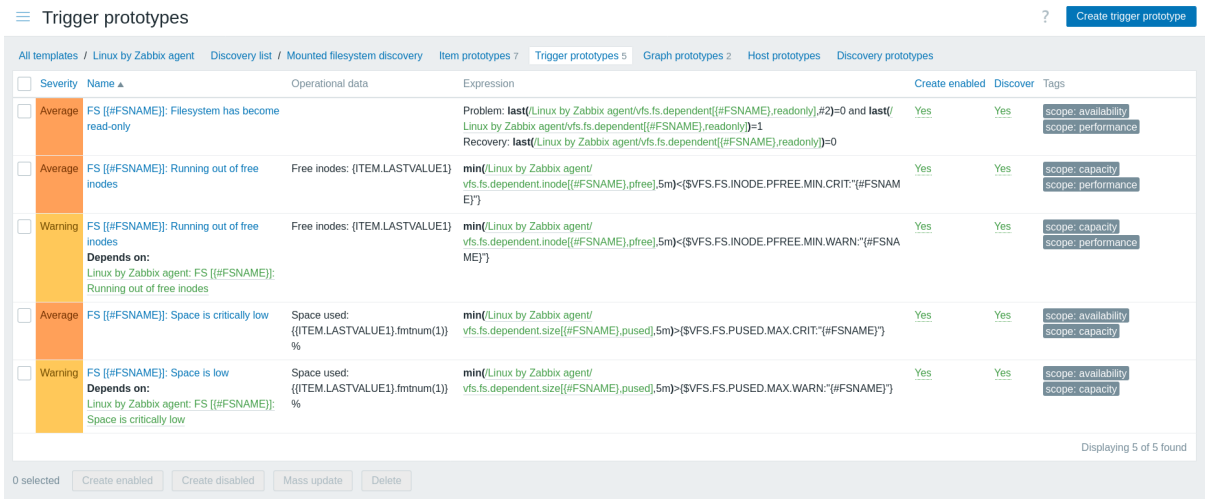
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami pozycji, a następnie kliknij wymagany przycisk.

Prototypy wyzwalaczy

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy wyzwalaczy reguły wykrywania niskiego poziomu na szablonie.

Jeśli szablon jest połączony z hostem, prototypy wyzwalaczy staną się podstawą do utworzenia rzeczywistych wyzwalaczy hosta podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Severity	Poziom ważności wyzwalacza jest wyświetlany zarówno za pomocą nazwy, jak i koloru tła komórki.
Name	Nazwa prototypu wyzwalacza wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wyzwalacza. Jeśli prototyp wyzwalacza należy do połączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wyzwalaczy na poziomie połączonego szablonu.
Expression	Wyświetlane jest wyrażenie wyzwalacza. Część wyrażenia dotycząca pozycji szablonu jest wyświetlana jako link prowadzący do formularza konfiguracji pozycji.
Operational data	Wyświetlany jest format danych operacyjnych wyzwalacza, zawierający dowolne ciągi znaków i makra, które zostaną dynamicznie rozwinięte w sekcji *Monitoring* > *Problems*.
Create enabled	Utworzenie wyzwalacza na podstawie tego prototypu jako: **Yes** - włączone **No** - wyłączone. Możesz przełączać się między wartościami „Yes” i „No”, klikając je.
Discover	Wykrywanie wyzwalacza na podstawie tego prototypu: **Yes** - wykrywaj **No** - nie wykrywaj. Możesz przełączać się między wartościami „Yes” i „No”, klikając je.
Tags	Wyświetlane są znaczniki prototypu wyzwalacza.

Aby skonfigurować nowy prototyp wyzwalacza, kliknij przycisk *Create trigger prototype* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz jako włączone* - utwórz te wyzwalacze jako *Włączone*
- *Utwórz jako wyłączone* - utwórz te wyzwalacze jako *Wyłączone*
- *Masowa aktualizacja* - masowo zaktualizuj te prototypy wyzwalaczy
- *Usuń* - usuń te prototypy wyzwalaczy

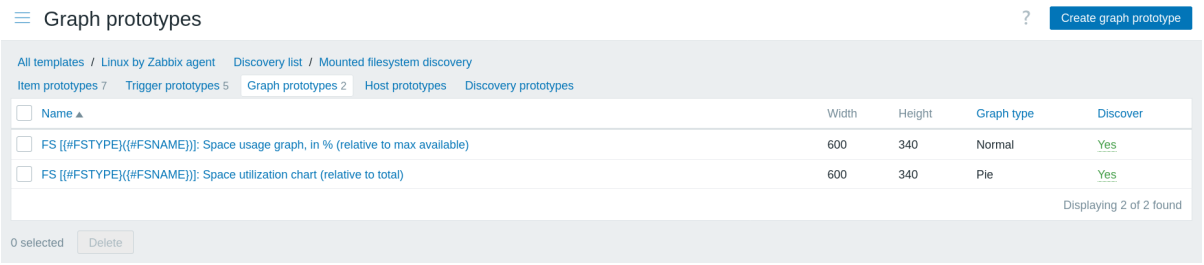
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami wyzwalaczy, a następnie kliknij wymagany przycisk.

Prototypy wykresów

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy wykresów reguły wykrywania niskopoziomowego w szablonie.

Jeśli szablon jest powiązany z hostem, prototypy wykresów staną się podstawą do tworzenia rzeczywistych wykresów hosta podczas wykrywania niskopoziomowego.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wykresu wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykresu. Jeśli prototyp wykresu należy do powiązanego szablonu, nazwa szablonu jest wyświetlana przed nazwą wykresu jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykresów na poziomie powiązanego szablonu.
Szerokość	Wyświetlana jest szerokość prototypu wykresu.
Wysokość	Wyświetlana jest wysokość prototypu wykresu.
Typ	Wyświetlany jest typ prototypu wykresu - *Normalny*, *Skumulowany*, *Kołowy* lub *Rozstrzelony*.

Kolumna	Opis
Wykrywaj	Wykrywanie wykresu na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Klikając te wartości, można przełączać się między „Tak” i „Nie”.

Aby skonfigurować nowy prototyp wykresu, kliknij przycisk *Utwórz prototyp wykresu* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Usuń* - usuń te prototypy wykresów

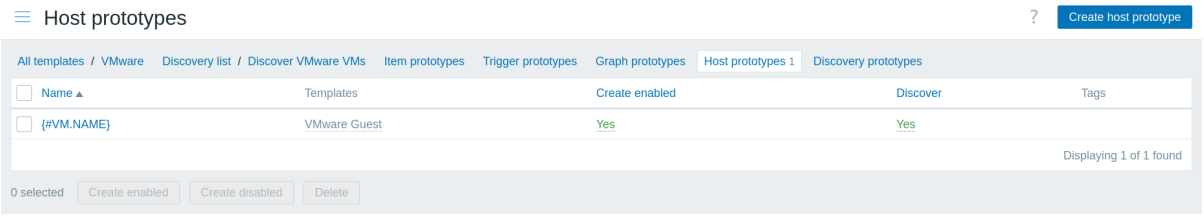
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami wykresów, a następnie kliknij wymagany przycisk.

Prototypy hostów

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy hostów reguły wykrywania niskopoziomowego w szablonie.

Jeśli szablon jest powiązany z hostem, prototypy hostów będą podstawą do tworzenia rzeczywistych hostów podczas wykrywania niskopoziomowego.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu hosta wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu hosta. Jeśli prototyp hosta należy do powiązanego szablonu, nazwa szablonu jest wyświetlana przed nazwą hosta jako szary link. Kliknięcie linku szablonu otwiera listę prototypów hostów na poziomie powiązanego szablonu.
Szablony	Wyświetlane są szablony prototypu hosta.
Tworzenie włączone	Utwórz host na podstawie tego prototypu jako: **Tak** - włączone **Nie** - wyłączone. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Wykrywanie	Wykryj host na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Znaczniki	Wyświetlane są znaczniki prototypu hosta.

Aby skonfigurować nowy prototyp hosta, kliknij przycisk *Utwórz prototyp hosta* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te hosty jako *Włączone*
- *Utwórz wyłączone* - utwórz te hosty jako *Wyłączone*
- *Usuń* - usuń te prototypy hostów

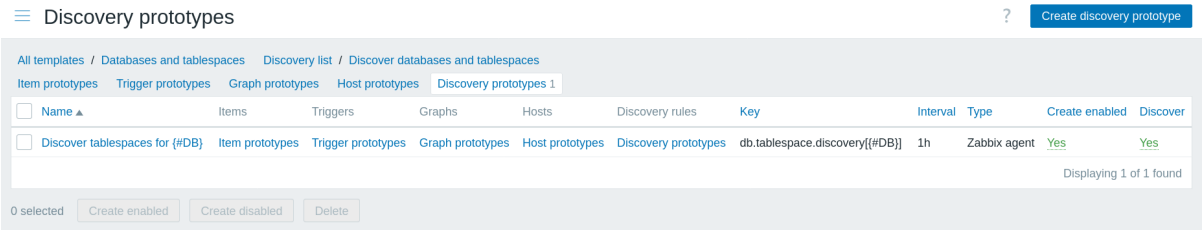
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami hostów, a następnie kliknij wymagany przycisk.

Prototypy wykrywania

Przegląd

W tej sekcji wyświetlane są skonfigurowane prototypy wykrywania reguły wykrywania niskopoziomowego na szablonie.

Jeśli szablon jest powiązany z hostem, prototypy wykrywania staną się podstawą do utworzenia rzeczywistych prototypów wykrywania nadrzędnej reguły wykrywania.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wykrywania wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykrywania. Jeśli prototyp wykrywania należy do powiązanego szablonu, nazwa szablonu jest wyświetlana przed nazwą prototypu wykrywania jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykrywania na poziomie powiązanego szablonu.
Pozycje	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Wyzwalacze	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Wykresy	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosty	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Reguły wykrywania	Wyświetlany jest link do listy prototypów wykrywania. Liczba istniejących prototypów wykrywania jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji używany do wykrywania.
Interwał	Wyświetlana jest częstotliwość wykonywania wykrywania.
Typ	Wyświetlany jest typ pozycji używany do wykrywania (agent Zabbix, agent SNMP itd.).
Tworzenie włączone	Wyświetlany jest status tworzenia prototypu wykrywania — tworzenie włączone (*Tak*) lub tworzenie wyłączone (*Nie*). Klikając status, można go zmienić z „Tak” na „Nie” (i odwrotnie).
Wykrywanie	Wyświetlany jest status wykrywania prototypu wykrywania — wykrywaj (*Tak*) lub nie wykrywaj (*Nie*). Klikając status, można go zmienić z „Tak” na „Nie” (i odwrotnie).

Aby skonfigurować nowy prototyp wykrywania, kliknij przycisk *Utwórz prototyp wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te prototypy wykrywania jako *Włączone*
- *Utwórz wyłączone* - utwórz te prototypy wykrywania jako *Wyłączone*
- *Usuń* - usuń te prototypy wykrywania

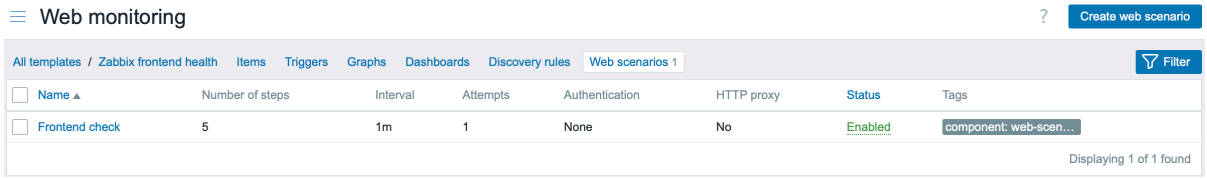
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami wykrywania, a następnie kliknij wymagany przycisk.

Scenariusze webowe

Przegląd

Lista scenariuszy internetowych dla szablonu jest dostępna w sekcji *Zbieranie danych* > *Szablony* po kliknięciu opcji *Web* przy odpowiednim szablonie.

Wyświetlana jest lista istniejących scenariuszy internetowych.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa scenariusza internetowego. Kliknięcie nazwy scenariusza internetowego otwiera formularz konfiguracji scenariusza internetowego. Jeśli scenariusz internetowy jest dziedziczony z innego szablonu, przed nazwą scenariusza internetowego wyświetlana jest nazwa szablonu jako szary link. Kliknięcie linku szablonu otwiera listę scenariuszy internetowych na poziomie tego szablonu.
Liczba kroków	Liczba kroków zawartych w scenariuszu.
Interwał aktualizacji	Częstotliwość wykonywania scenariusza.
Próby	Liczba prób wykonania kroków scenariusza internetowego.
Uwierzytelnianie	Wyświetlana jest metoda uwierzytelniania: Basic, NTLM, Kerberos, Digest lub None.
Proxy HTTP	Wyświetlany jest proxy HTTP lub „No”, jeśli nie jest używany.
Stan	Wyświetlany jest stan scenariusza internetowego: *Enabled* lub *Disabled*. Kliknięcie stanu umożliwia jego zmianę.
Znaczniki	Wyświetlane są znaczniki scenariusza internetowego. Można wyświetlić maksymalnie trzy znaczniki (pary nazwa:wartość). Jeśli znaczników jest więcej, wyświetlany jest link „...”, który umożliwia wyświetlenie wszystkich znaczników po najechaniu kursorem myszy.

Aby skonfigurować nowy scenariusz internetowy, kliknij przycisk *Create web scenario* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Włącz* - zmień status scenariusza na *Włączony*
- *Wyłącz* - zmień status scenariusza na *Wyłączony*
- *Usuń* - usuń scenariusze web

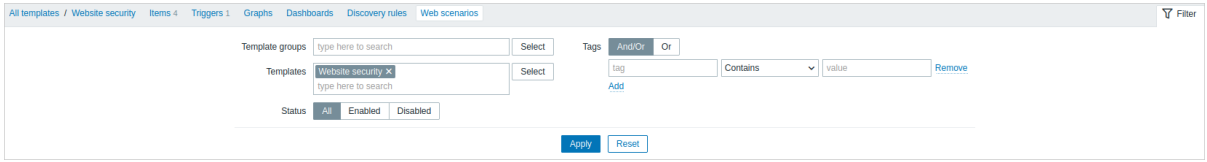
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi scenariuszami web, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te scenariusze, które Cię interesują.

Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane przy nierozwiązanych makrach.

Łącze *Filter* jest dostępne nad listą scenariuszy WWW. Po kliknięciu pojawia się filtr, za pomocą którego można filtrować scenariusze według grupy szablonów, szablonu, statusu i tagów.

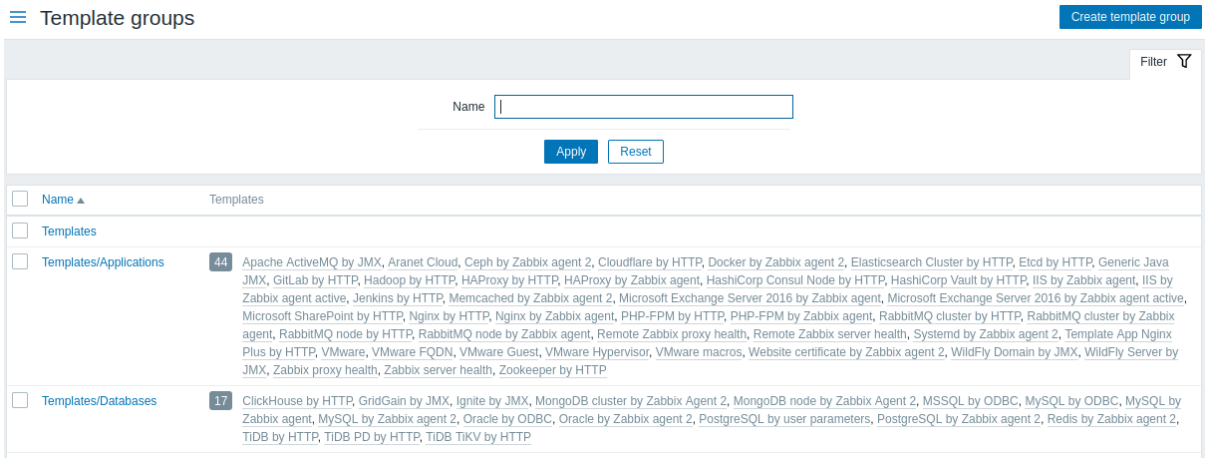


Grupy szablonów

Przegląd

W sekcji *Zbieranie danych* > *Grupy szablonów* użytkownicy mogą konfigurować i zarządzać grupami szablonów.

Wyświetlana jest lista istniejących grup szablonów wraz z ich szczegółami. Można wyszukiwać i filtrować grupy szablonów według nazwy.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy szablonów. Kliknięcie nazwy grupy otwiera formularz konfiguracji grupy szablonów.
Szablony	Liczba szablonów w grupie (wyświetlana na szaro), a następnie lista członków grupy. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu. Kliknięcie liczby otwiera listę szablonów w tej grupie.

Opcje masowej edycji

Aby usunąć kilka szablonów jednocześnie, zaznacz pola wyboru przed odpowiednimi szablonami, a następnie kliknij przycisk Delete poniżej listy.

Używanie filtra

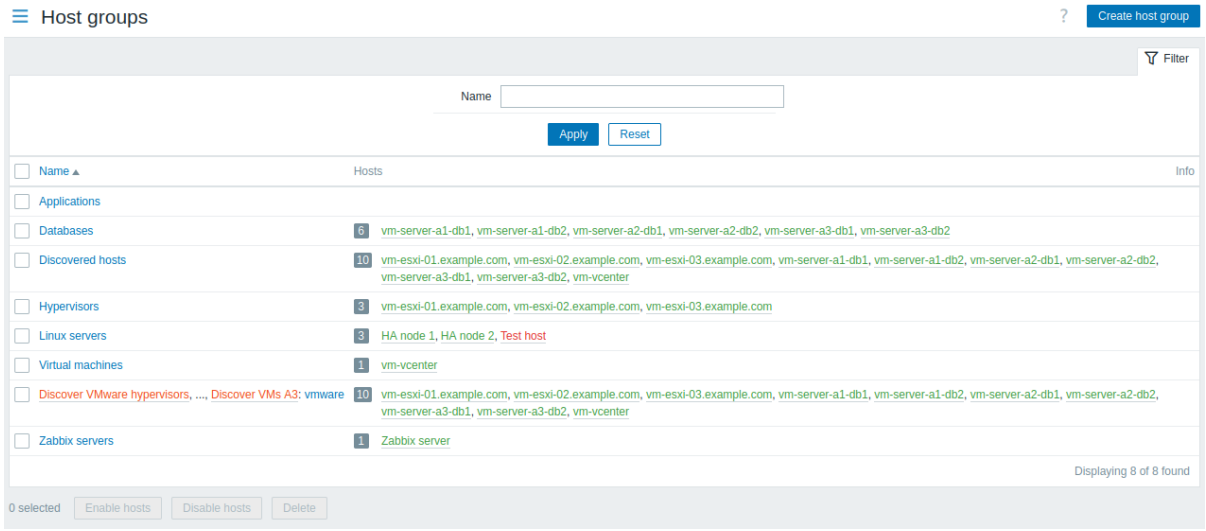
Możesz użyć filtra, aby wyświetlić tylko te grupy szablonów, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane przy nierozwiniętych makrach.

Grupy hostów

Przegląd

W sekcji *Zbieranie danych* > *Grupy hostów* użytkownicy mogą konfigurować i zarządzać grupami hostów.

Wyświetlana jest lista istniejących grup hostów wraz ze szczegółowymi informacjami. Można wyszukiwać i filtrować grupy hostów według nazwy.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy hostów. Kliknięcie nazwy grupy otwiera formularz konfiguracji grupy hostów. Wykryte grupy hostów są wyświetlane z nazwami reguł wykrywania niskopoziomowego jako prefiksem. Kliknięcie nazwy reguły LLD otwiera formularz konfiguracji prototypu hosta. Należy pamiętać, że wykryte grupy hostów są usuwane po usunięciu wszystkich reguł LLD, które je wykryły.
Hosty	Liczba hostów w grupie (wyświetlana na szaro), a następnie lista członków grupy. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Kliknięcie liczby spowoduje odfiltrowanie, z całej listy hostów, tych, które należą do grupy.
Informacje	Wyświetlane są informacje o błędach (jeśli występują) dotyczące grupy hostów.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Włącz hosty* - zmień status wszystkich hostów w grupie na ”Monitorowane”
- *Wyłącz hosty* - zmień status wszystkich hostów w grupie na ”Nie monitorowane”
- *Usuń* - usuń grupy hostów

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi grupami hostów, a następnie kliknij wymagany przycisk.

Używanie filtra

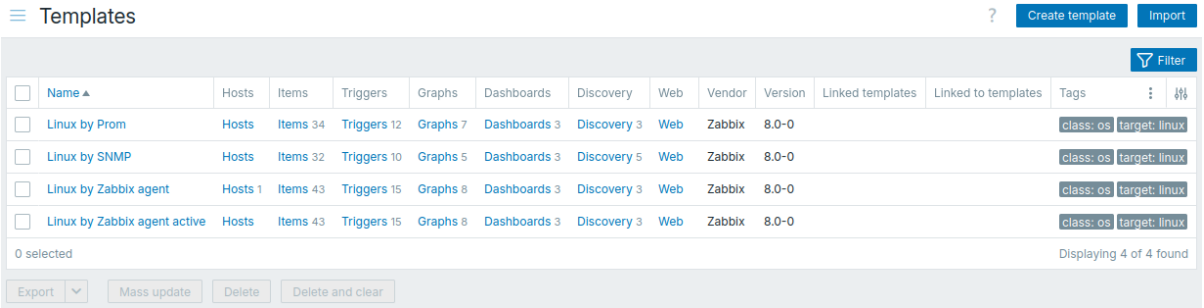
Możesz użyć filtra, aby wyświetlić tylko te grupy hostów, którymi jesteś zainteresowany. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

Szablony

Przegląd

W sekcji *Gromadzenie danych* > *Szablony* użytkownicy mogą konfigurować i utrzymywać szablony.

Wyświetlana jest lista istniejących szablonów wraz ze szczegółowymi informacjami.



Szczegóły szablonów są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia wyświetlanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie domyślnego układu. Można również zmieniać szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślną szerokość. Wszystkie dostosowania są stosowane natychmiast i zawsze zapisywane dla bieżącego użytkownika.

Dostępne są następujące kolumny:

Kolumna	Opis
Nazwa	Nazwa szablonu. Kliknięcie nazwy szablonu otwiera formularz konfiguracji szablonu.
Hosty	Liczba edytowalnych hostów, z którymi połączono szablon; hosty tylko do odczytu nie są uwzględniane. Kliknięcie *Hosty* otwiera listę hostów, na której filtrowane są tylko hosty połączone z szablonem.
Jednostki (pozycje, wyzwalacze, wykresy, pulpity nawigacyjne, wykrywanie, Web)	Liczba odpowiednich jednostek w szablonie (wyświetlana na szaro). Kliknięcie nazwy jednostki powoduje odfiltrowanie w całej liście tej jednostki elementów należących do szablonu.
Połączone szablony	Szablony, które są połączone z szablonem.
Połączono z szablonami	Szablony, z którymi szablon jest połączony.

Kolumna	Opis
Dostawca, *Wersja*	Dostawca i wersja szablonu; wyświetlane, jeśli konfiguracja szablonu zawiera takie informacje, i tylko w przypadku gotowych szablonów, zaimportowanych szablonów lub szablonów zmodyfikowanych za pośrednictwem Template API. W przypadku gotowych szablonów wersja jest wyświetlana w następujący sposób: główna wersja Zabbix, separator (”-”), numer rewizji (zwiększany przy każdej nowej wersji szablonu i resetowany przy każdej głównej wersji Zabbix). Na przykład: 7.0-0, 7.0-3, 8.0-0, 8.0-3.
Znaczniki	Wyświetlane są znaczniki szablonu, bez rozwiniętych makr. Umieszczenie wskaźnika myszy nad ikoną lub kliknięcie ikony trzech kropek za wyświetlanymi znacznikami otwiera wyskakujące okno ze wszystkimi znacznikami. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia ustawienie liczby wyświetlanych znaczników (1–3), wybór sposobu wyświetlania nazw znaczników (pełne, skrócone lub brak), ustawienie priorytetu wyświetlania znaczników albo zduplikowanie kolumny (zduplikowane kolumny można zmienić lub usunąć).
Wartość znacznika	Wyświetlane są wartości znaczników szablonu, bez rozwiniętych makr. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia zmianę nazwy kolumny, wybór wyświetlanych wartości znaczników albo zduplikowanie kolumny (zduplikowane kolumny można zmienić lub usunąć).
Tekst niestandardowy	Wyświetlany jest tekst niestandardowy, wartości makr lub połączenie obu tych elementów. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia zmianę nazwy kolumny, wprowadzenie tekstu niestandardowego lub makr albo zduplikowanie kolumny (zduplikowane kolumny można zmienić lub usunąć). Obsługiwane makra: `{HOST.ID}`, `{HOST.HOST}`, `{HOST.NAME}`, `{HOST.DESCRIPTION}` oraz makra użytkownika.

Aby skonfigurować nowy szablon, kliknij przycisk *Utwórz szablon* w prawym górnym rogu.

Aby zaimportować szablon z pliku YAML, XML lub JSON, kliknij przycisk *Importuj* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

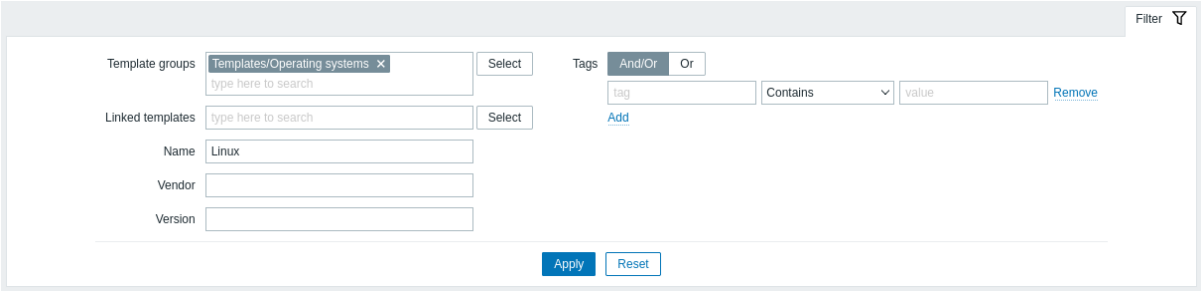
- *Eksport* - eksport szablonu do pliku YAML, XML lub JSON;
- *Masowa aktualizacja* - aktualizacja kilku właściwości dla wielu szablonów jednocześnie;
- *Usuń* - usuń szablon, pozostawiając jego powiązane obiekty (pozycje, wyzwalacze itp.) na hostach;
- *Usuń i wyczyść* - usuń szablon i jego powiązane obiekty z hostów.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi szablonami, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te szablony, które Cię interesują. Aby poprawić wydajność wyszukiwania, dane są przeszukiwane z nierozwiniętymi makrami.

Link *Filter* jest dostępny poniżej przycisków *Create template* i *Import*. Po kliknięciu staje się dostępny filtr, za pomocą którego można filtrować szablony według grup szablonów, powiązanych szablonów, nazwy i tagów.



Parameter	Description
Template groups	Filtrowanie według jednej lub kilku grup szablonów. Określenie nadrzędnej grupy szablonów automatycznie zaznacza wszystkie zagnieżdżone grupy.
Linked templates	Filtrowanie według bezpośrednio powiązanych szablonów.
Name	Filtrowanie według nazwy szablonu.
Vendor	Filtrowanie według dostawcy szablonu.
Version	Filtrowanie według wersji szablonu.

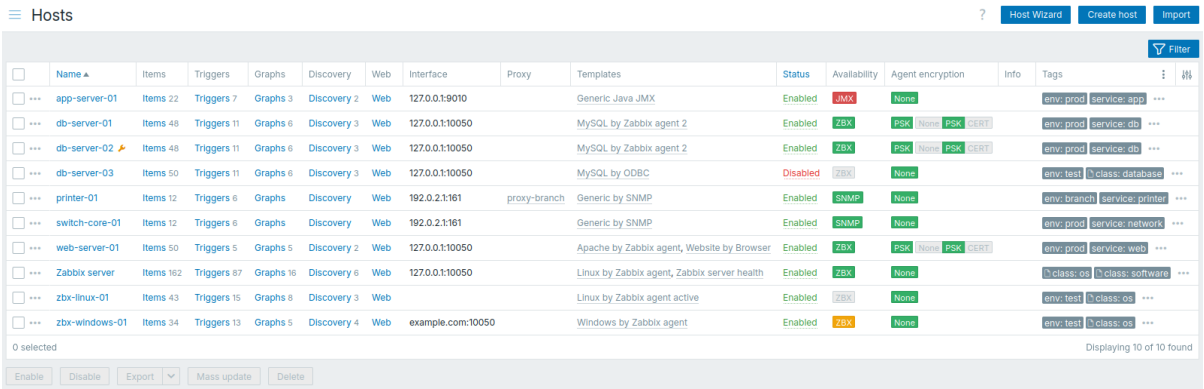
Parameter	Description
Tags	Filtrowanie według nazwy i wartości tagu szablonu. Filtrowanie jest możliwe tylko dla tagów na poziomie szablonu (nie dziedziczonych). Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżnianie wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżnianie wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - muszą zostać spełnione wszystkie warunki, warunki o tej samej nazwie tagu będą grupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek.

Hosty

Przegląd

W sekcji *Zbieranie danych* > *Hosty* użytkownicy mogą konfigurować i zarządzać hostami.

Wyświetlana jest lista istniejących hostów wraz ze szczegółami.



Szczegóły hostów są wyświetlane w konfigurowalnej tabeli. Przycisk *Dostosuj tabelę* w nagłówku tabeli umożliwia wyświetlanie, ukrywanie lub zmianę kolejności kolumn oraz przywrócenie domyślnego układu. Można również zmieniać szerokość każdej kolumny, przeciągając jej prawą krawędź; dwukrotne kliknięcie krawędzi przywraca domyślną szerokość. Wszystkie dostosowania są stosowane natychmiast i zawsze zapamiętywane dla bieżącego użytkownika.

Dostępne są następujące kolumny:

Kolumna	Opis
Nazwa	Nazwa hosta. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Pomarańczowa ikona klucza za nazwą hosta oznacza, że ten host jest w trybie konserwacji. Szczegóły konserwacji są wyświetlane po umieszczeniu wskaźnika myszy na ikonie.
Jednostki (pozycje, wyzwalacze, wykresy, wykrywanie, Web)	Kliknięcie nazwy jednostki wyświetla pozycje, wyzwalacze itd. hosta. Liczba odpowiednich jednostek jest wyświetlana na szaro.
Interfejs	Wyświetlany jest główny interfejs hosta.

Kolumna	Opis
Proxy	W tej kolumnie wyświetlane są przypisane proxy: <Proxy name> - host jest monitorowany przez niezależny proxy (nawet jeśli proxy jest częścią grupy proxy); <Proxy group name: proxy name> - host jest monitorowany przez grupę proxy, a serwer Zabbix przypisał proxy do monitorowania hosta; <Proxy group name> - host jest monitorowany przez grupę proxy bez żadnych proxy lub serwer Zabbix nie przypisał proxy do monitorowania hosta; Brak - host nie jest monitorowany ani przez proxy, ani przez grupę proxy. Ta kolumna jest wyświetlana tylko wtedy, gdy opcja filtra *Monitorowany przez* jest ustawiona na „Dowolny", „Proxy" lub „Grupa proxy".
Szablony	Wyświetlane są szablony powiązane z hostem. Jeśli w powiązanym szablonie zawarte są inne szablony, są one wyświetlane w nawiasach i oddzielone przecinkami. Kliknięcie nazwy szablonu otwiera jego formularz konfiguracji.
Stan	Wyświetlany jest stan hosta - *Włączony* lub *Wyłączony*. Klikając stan, można zmienić go ręcznie. Wykryte hosty, które zostały utracone, są oznaczone ikoną informacji. Tekst podpowiedzi zawiera szczegółowe informacje o ich stanie.
Dostępność	Wyświetlana jest dostępność hosta dla każdego skonfigurowanego interfejsu. Ikony dostępności przedstawiają bieżący stan interfejsu hosta na serwerze Zabbix. Dlatego jeśli wyłączysz hosta w frontend, jego dostępność zostanie zaktualizowana po zsynchronizowaniu przez serwer Zabbix zmian konfiguracji. Podobnie, jeśli włączysz hosta, jego dostępność zostanie zaktualizowana po zsynchronizowaniu przez serwer Zabbix zmian konfiguracji i odpytaniu hosta. Ikony dostępności przedstawiają tylko te typy interfejsów (Agent, SNMP, IPMI, JMX), które są skonfigurowane. Umieszczenie wskaźnika myszy na ikonie wyświetla wyskakujące okno z listą wszystkich interfejsów tego samego typu wraz ze szczegółami, stanem i błędami. W przypadku interfejsu Agent wyskakujące okno wyświetla interfejsy (pasywne) i aktywne sprawdzania. Jeśli host ma tylko aktywne sprawdzania, wyświetlana jest ikona interfejsu Agent, nawet jeśli host nie ma skonfigurowanego interfejsu Agent. Kolumna jest pusta dla hostów bez interfejsów. Stan pojedynczego interfejsu hosta jest określany na podstawie połączenia między włączoną pozycją korzystającą z interfejsu a hostem. Stan może być następujący: **Dostępny** - połączenie z hostem zakończyło się powodzeniem; **Niedostępny** - połączenie z hostem nie powiodło się (przekroczenie limitu czasu, problemy z zaporą sieciową itd.); **Nieznany** - nie podjęto próby połączenia z hostem lub wynik jest nieznany. Dodatkowe informacje o sposobie określania stanu interfejsu przez serwer Zabbix zawiera sekcja Nieznany stan interfejsu oraz Ustawienia nieosiągalnego/niedostępnego interfejsu hosta. Stan wszystkich interfejsów hosta danego typu (Agent, SNMP, IPMI, JMX) jest określany na podstawie tych interfejsów, z których korzysta co najmniej jedna włączona pozycja. Stan jest wskazywany kolorem ikony: **Zielony** - wszystkie interfejsy są dostępne; **Żółty** - co najmniej jeden interfejs jest niedostępny, a co najmniej jeden jest dostępny lub ma stan nieznany; **Czerwony** - wszystkie interfejsy są niedostępne; **Szary** - co najmniej jeden interfejs ma stan nieznany, ale żaden nie jest niedostępny. **Dostępność aktywnych sprawdzań.** Jeśli na hoście włączone jest co najmniej jedno aktywne sprawdzanie, aktywne sprawdzania wpływają również na całkowitą dostępność interfejsu Agent zgodnie z powyższym opisem. Aby określić dostępność aktywnych sprawdzań, w wątku aktywnych sprawdzań agenta wysyłane są komunikaty heartbeat. Częstotliwość komunikatów heartbeat jest kontrolowana przez parametr `HeartbeatFrequency` w konfiguracji agenta Zabbix lub agenta 2 (domyślnie 60 sekund, zakres 0-3600). Aktywne sprawdzania są uznawane za niedostępne, gdy komunikat heartbeat aktywnego sprawdzania jest starszy niż 2 x HeartbeatFrequency sekund. **Uwaga:** agenty Zabbix starsze niż w wersji 6.2.x nie wysyłają komunikatów heartbeat aktywnych sprawdzań, dlatego dostępność ich hostów pozostaje nieznana.

Kolumna	Opis
Szyfrowanie agenta	Wyświetlany jest stan szyfrowania połączeń z hostem i z hosta: **Brak** - brak szyfrowania; **PSK** - używany jest klucz wstępnie współdzielony; **Cert** - używany jest certyfikat.
Informacje	Wyświetlane są informacje o błędach (jeśli występują) dotyczących hosta.
Tagi	Wyświetlane są tagi hosta, bez rozwiniętych makr. Umieszczenie wskaźnika myszy na ikonie trzech kropek lub kliknięcie jej po wyświetlonych tagach otwiera wyskakujące okno ze wszystkimi tagami. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia ustawienie liczby wyświetlanych tagów (1–3), wybór sposobu wyświetlania nazw tagów (pełne, skrócone lub brak), ustawienie priorytetu wyświetlania tagów albo zduplikowanie kolumny (zduplikowane kolumny można zmienić lub usunąć).
Wartość tagu	Wyświetlane są wartości tagów hosta, bez rozwiniętych makr. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia zmianę nazwy kolumny, wybór wyświetlanych wartości tagów albo zduplikowanie kolumny (zduplikowane kolumny można zmienić lub usunąć).
Tekst niestandardowy	Wyświetlany jest tekst niestandardowy, wartości makr lub połączenie obu tych elementów. Kliknięcie przycisku *Opcje* w nagłówku kolumny umożliwia zmianę nazwy kolumny, wprowadzenie tekstu niestandardowego lub makr albo zduplikowanie kolumny (zduplikowane kolumny można zmienić lub usunąć). Obsługiwane makra: `{HOST.*}` (z wyjątkiem `{HOST.METADATA}`), `{INVENTORY.*}` oraz makra użytkownika.

Aby otworzyć menu hosta, kliknij przycisk *Opcje* przed nazwą hosta.

Aby skonfigurować nowego hosta, kliknij przycisk *Kreator hosta* lub *Utwórz hosta* w prawym górnym rogu.

Aby zaimportować hosta z pliku YAML, XML lub JSON, kliknij przycisk *Importuj* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

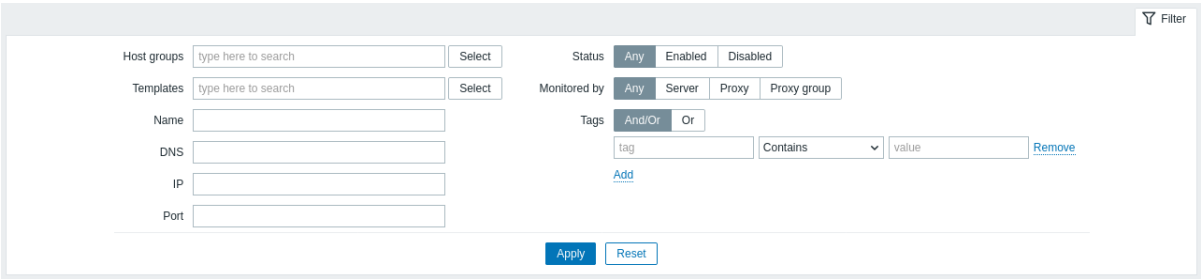
- *Enable* - zmień status hosta na *Monitored*;
- *Disable* - zmień status hosta na *Not monitored*;
- *Export* - eksportuj hosty do pliku YAML, XML lub JSON;
- *Mass update* - zaktualizuj kilka właściwości dla wielu hostów jednocześnie;
- *Delete* - usuń hosty.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi hostami, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko hosty, które Cię interesują. Aby poprawić wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym możesz określić żądane kryteria filtrowania.



Parameter	Description
Host groups	Filtrowanie według jednej lub kilku grup hostów. Określenie nadrzędnej grupy hostów domyślnie zaznacza wszystkie zagnieżdżone grupy hostów.
Templates	Filtrowanie według powiązanych szablonów.
Name	Filtrowanie według widocznej nazwy hosta.
DNS	Filtrowanie według nazwy DNS.
IP	Filtrowanie według adresu IP.

Parameter	Description
Port	Filtrowanie według numeru portu.
Status	Filtrowanie według statusu hosta.
Monitored by	Filtrowanie hostów monitorowanych przez serwer Zabbix, proxy lub grupę proxy.
Proxies	Filtrowanie hostów monitorowanych przez wskazane tutaj proxy. To pole jest dostępne tylko wtedy, gdy w polu *Monitored by* wybrano opcję ”Proxy”.
Proxy groups	Filtrowanie hostów monitorowanych przez wskazane tutaj grupy proxy. To pole jest dostępne tylko wtedy, gdy w polu *Monitored by* wybrano opcję ”Proxy group”.
Tags	Filtrowanie według nazwy i wartości tagu hosta. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek.

Nieznany status interfejsu

serwer Zabbix wyświetla status ”Unknown” dla interfejsu hosta (Agent, SNMP, IPMI, JMX) w następujących przypadkach:

- host jest wyłączony.
- host jest ustawiony do monitorowania przez proxy, inne proxy lub serwer, jeśli wcześniej był monitorowany przez proxy.
- host jest monitorowany przez proxy, które wydaje się być offline (brak aktualizacji otrzymanych z proxy w ciągu maksymalnego interwału heartbeat - 1 godzina).
- Wszystkie pozycje hosta z tym typem interfejsu są wyłączone.
- Nie skonfigurowano żadnych pollerów dla tego typu interfejsu (na przykład parametr konfiguracyjny serwera `StartAgentPollers` lub `StartSNMPPollers` jest ustawiony na 0).

Dostępność interfejsu jest ustawiana na ”Unknown” po synchronizacji cache konfiguracji serwer Zabbix.

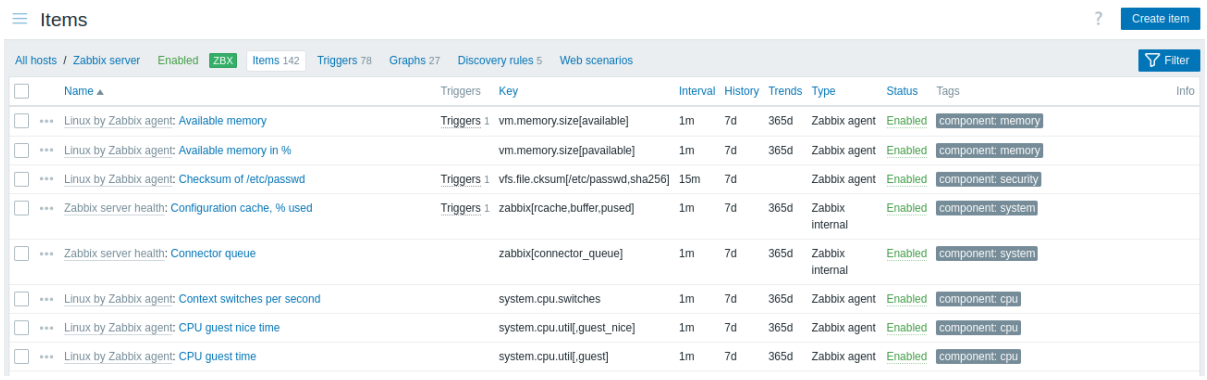
Dostępność interfejsu (Available/Not available) na hostach monitorowanych przez proxy jest przywracana po synchronizacji cache konfiguracji proxy.

Pozycje

Przegląd

Lista pozycji dla hosta jest dostępna w sekcji *Zbieranie danych* > *Hosty* po kliknięciu opcji *Pozycje* przy odpowiednim hoście.

Wyświetlana jest lista istniejących pozycji.



Wyświetlane dane:

Kolumna	Opis
Menu pozycji	Kliknij ikonę z trzema kropkami, aby otworzyć menu pozycji.
Host	Host pozycji. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele hostów lub nie wybrano żadnego hosta.
Nazwa	Nazwa pozycji wyświetlana jako niebieski odnośnik do szczegółów pozycji. Kliknięcie odnośnika z nazwą pozycji otwiera formularz konfiguracji pozycji. Jeśli pozycja hosta należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą pozycji jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę pozycji na poziomie szablonu. Jeśli pozycja została utworzona na podstawie prototypu pozycji, jej nazwa jest poprzedzona nazwą reguły wykrywania niskopoziomowego, wyświetlaną na pomarańczowo. Kliknięcie nazwy reguły wykrywania otwiera listę prototypów pozycji.
Wyzwalacze	Umieszczenie wskaźnika myszy nad sekcją Wyzwalacze wyświetla okno informacyjne z wyzwalaczami powiązanymi z pozycją. Liczba wyzwalaczy jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania. *Uwaga*: pozycje pasywne można również sprawdzić natychmiast, naciskając przycisk *Wykonaj teraz*.
Historia	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trendy	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Typ	Wyświetlany jest typ pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itd.).
Stan	Wyświetlany jest stan pozycji: *Włączona*, *Wyłączona* lub *Nieobsługiwana*. Stan można zmienić ręcznie, klikając go: z Włączona na Wyłączona (i odwrotnie) oraz z Nieobsługiwana na Wyłączona (i odwrotnie). Wykryte pozycje, które zostały utracone, są oznaczone ikoną informacji. Tekst etykiety narzędzia zawiera szczegółowe informacje o ich stanie.
Znaczniki	Wyświetlane są znaczniki pozycji. Można wyświetlić maksymalnie trzy znaczniki (pary nazwa:wartość). Jeśli znaczników jest więcej, wyświetlany jest odnośnik „...", który umożliwia wyświetlenie wszystkich znaczników po umieszczeniu nad nim wskaźnika myszy.
Informacje	Jeśli pozycja działa prawidłowo, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i". Umieść wskaźnik myszy nad ikoną, aby wyświetlić etykietę narzędzia z opisem błędu.

Aby skonfigurować nową pozycję, kliknij przycisk *Utwórz pozycję* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

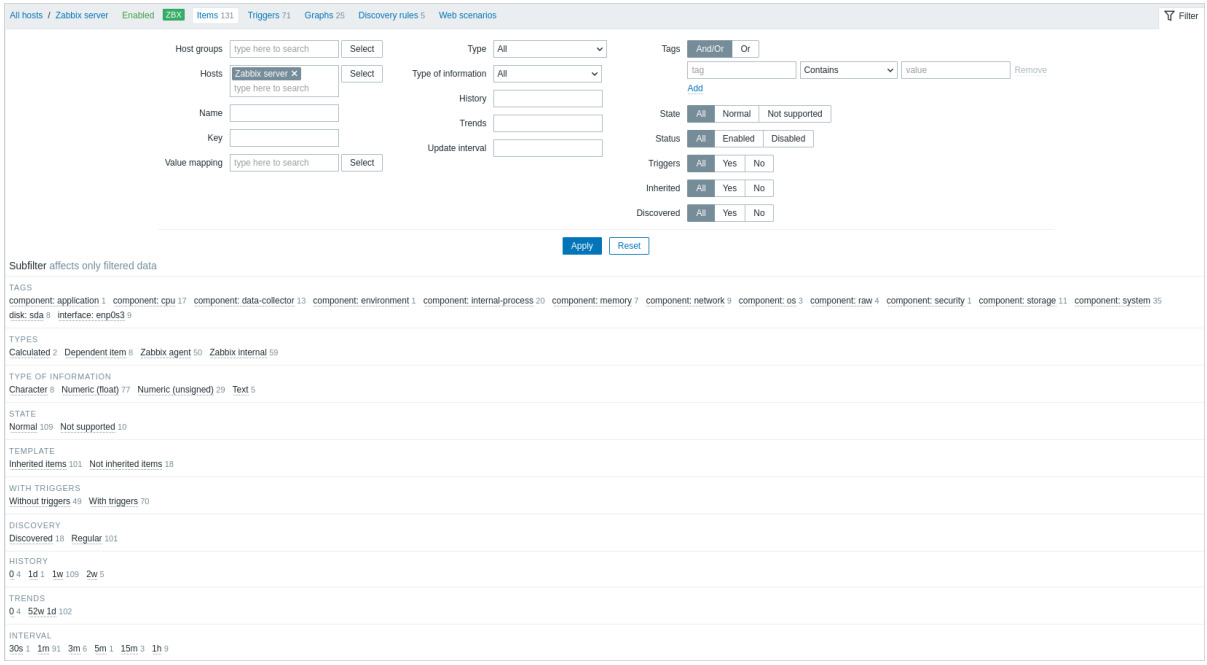
- *Enable* - zmień status pozycji na *Enabled*
- *Disable* - zmień status pozycji na *Disabled*
- *Execute now* - natychmiast wykonaj sprawdzenie nowych wartości pozycji. Obsługiwane tylko dla sprawdzeń **passive** (zobacz więcej szczegółów). Należy pamiętać, że podczas natychmiastowego sprawdzania wartości pamięć podręczna konfiguracji nie jest aktualizowana, więc wartości nie będą odzwierciedlać bardzo niedawnych zmian w konfiguracji pozycji.
- *Clear history and trends* - usuń dane historyczne i trendów dla pozycji.
- *Copy* - skopiuj pozycje do innych hostów lub szablonów.
- *Mass update* - zaktualizuj kilka właściwości dla wielu pozycji jednocześnie.
- *Delete* - usuń pozycje.

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich pozycjach, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko interesujące Cię pozycje. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym można określić żądane kryteria filtrowania.

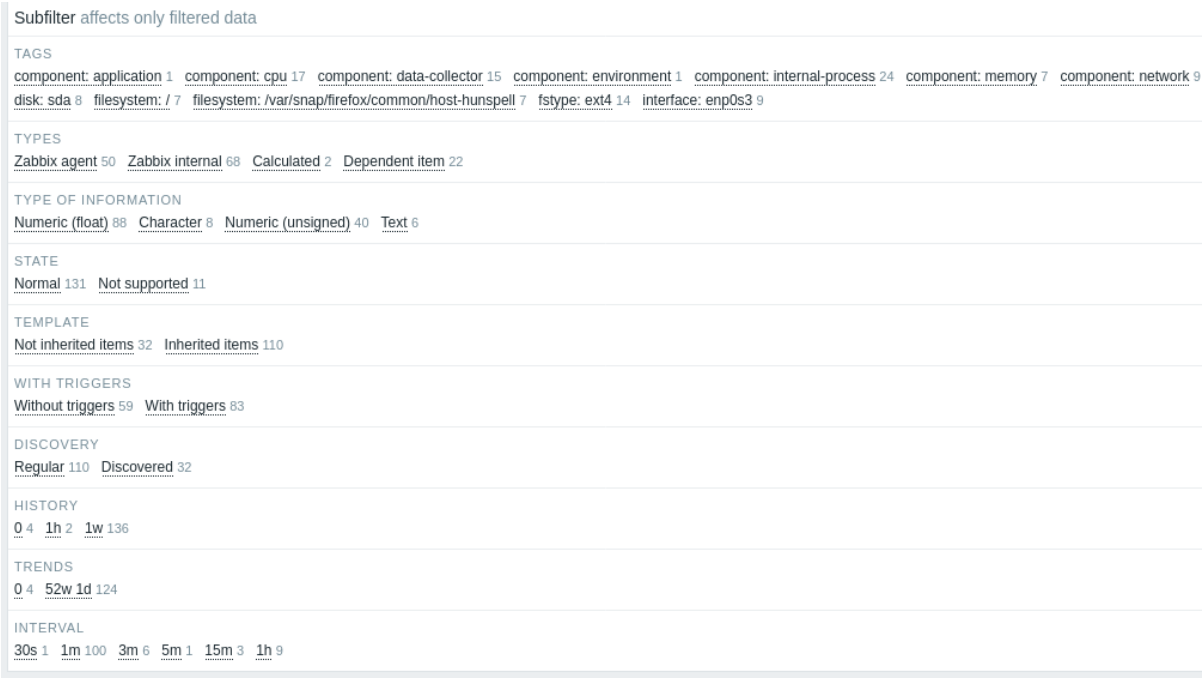


Parameter	Description
Host groups	Filtrowanie według jednej lub wielu grup hostów. Wskazanie nadrzędnej grupy hostów automatycznie zaznacza wszystkie zagnieżdżone grupy hostów. Nie można wybrać grup hostów zawierających wyłącznie szablony.
Hosts	Filtrowanie według jednego lub wielu hostów.
Name	Filtrowanie według nazwy pozycji.
Key	Filtrowanie według klucza pozycji.
Value mapping	Filtrowanie według użytego mapowania wartości. Ten parametr nie jest wyświetlany, jeśli opcja *Hosts* jest pusta.
Type	Filtrowanie według typu pozycji (Zabbix agent, SNMP agent itp.).
Type of information	Filtrowanie według typu informacji (Numeric unsigned, float itp.).
History	Filtrowanie według czasu przechowywania historii pozycji.
Trends	Filtrowanie według czasu przechowywania trendów pozycji.
Update interval	Filtrowanie według interwału aktualizacji pozycji.
Tags	Określ tagi, aby ograniczyć liczbę wyświetlanych pozycji. Można uwzględniać, jak i wykluczać określone tagi oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (rozróżniana wielkość liter) **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (rozróżniana wielkość liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek
State	Filtrowanie według stanu pozycji - *Normal* lub *Not supported*.
Status	Filtrowanie według statusu pozycji - *Enabled* lub *Disabled*.
Triggers	Filtrowanie pozycji z wyzwalaczami lub bez wyzwalaczy.
Inherited	Filtrowanie pozycji dziedziczonych lub niedziedziczonych z szablonu.
Discovery	Filtrowanie pozycji wykrytych lub niewykrytych przez niskopoziomowe wykrywanie.

Korzystanie z podfiltra

Podfiltr umożliwia dalszą modyfikację filtrowania zastosowanego przez filtr główny.

Zawiera klikalne odnośniki zapewniające szybki dostęp do powiązanych pozycji. Pozycje są powiązane wspólną cechą, taką jak znacznik, typ pozycji, stan pozycji, interwał aktualizacji pozycji itp. Po kliknięciu cechy zostaje ona wyróżniona szarym tłem, a pozycje są natychmiast filtrowane (nie trzeba klikać przycisku *Zastosuj* w filtrze głównym). Kliknięcie kolejnej cechy dodaje ją do przefiltrowanych wyników. Ponowne kliknięcie cechy usuwa filtrowanie.



Podfiltry są generowane na podstawie przefiltrowanych danych, które są ograniczone do 1000 rekordów. Aby wyświetlić więcej rekordów w podfiltrze, należy zwiększyć wartość parametru *Limit for search and filter results* (w sekcji *Administration* > *General* > *GUI*).

W przeciwieństwie do filtra głównego podfiltr jest aktualizowany przy każdym żądaniu odświeżenia tabeli, aby zawsze zawierać aktualne informacje o dostępnych opcjach filtrowania i ich licznikach.

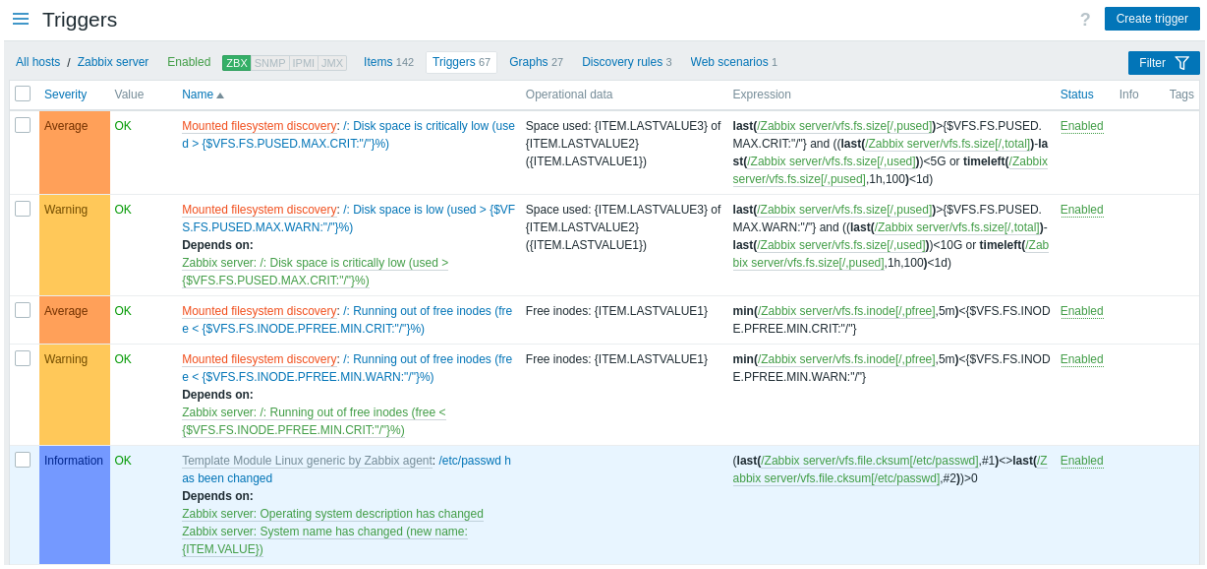
Liczba wyświetlanych cech jest ograniczona do 100 w poziomie. Jeśli jest ich więcej, na końcu wyświetlana jest ikona trzech kropek; nie można jej kliknąć.

Liczba obok każdej klikalnej cechy wskazuje liczbę pogrupowanych w niej pozycji (na podstawie wyników filtra głównego). Po kliknięciu cechy liczby przy innych dostępnych cechach są wyświetlane ze znakiem plus, wskazując, ile pozycji można dodać do bieżącego wyboru. Cechy bez pozycji nie są wyświetlane, chyba że zostały wcześniej wybrane w podfiltrze.

Wyzwalacze

Przegląd

Lista wyzwalaczy dla hosta jest dostępna w sekcji *Zbieranie danych* > *Hosty* po kliknięciu opcji *Wyzwalacze* przy odpowiednim hoście.



Wyświetlane dane:

Kolumna	Opis
Ważność	Ważność wyzwalacza jest wyświetlana zarówno za pomocą nazwy, jak i koloru tła komórki.
Wartość	Wyświetlany jest stan wyzwalacza: **OK** - stan OK **PROBLEM** - stan problemu
Host	Host wyzwalacza. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele hostów lub nie wybrano żadnego hosta.
Nazwa	Nazwa wyzwalacza wyświetlana jako niebieski odnośnik do szczegółów wyzwalacza. Kliknięcie odnośnika z nazwą wyzwalacza otwiera formularz konfiguracji wyzwalacza. Jeśli wyzwalacz hosta należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę wyzwalaczy na poziomie szablonu. Jeśli wyzwalacz został utworzony na podstawie prototypu wyzwalacza, jego nazwa jest poprzedzona nazwą reguły wykrywania niskopoziomowego, wyświetlaną na pomarańczowo. Kliknięcie nazwy reguły wykrywania otwiera listę prototypów wyzwalaczy.
Dane operacyjne	Definicja danych operacyjnych wyzwalacza zawierająca dowolne ciągi znaków i makra, które zostaną dynamicznie rozwinięte w sekcji *Monitorowanie* > *Problemy*.
Wyrażenie	Wyświetlane jest wyrażenie wyzwalacza. Część wyrażenia dotycząca hosta i pozycji jest wyświetlana jako odnośnik prowadzący do formularza konfiguracji pozycji.
Stan	Wyświetlany jest stan wyzwalacza - *Włączony*, *Wyłączony* lub *Nieznany*. Klikając stan, można zmienić go ręcznie - z Włączony na Wyłączony (i odwrotnie) oraz z Nieznany na Wyłączony (i odwrotnie). Problemy wyłączonego wyzwalacza nie są już wyświetlane we frontendzie, ale nie są usuwane. Wykryte wyzwalacze, które zostały utracone, są oznaczone ikoną informacji. Tekst etykiety narzędzia zawiera szczegółowe informacje o ich stanie.
Informacje	Jeśli wszystko działa prawidłowo, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i”. Umieść wskaźnik myszy nad ikoną, aby wyświetlić etykietę narzędzia z opisem błędu.
Znaczniki	Jeśli wyzwalacz zawiera znaczniki, w tej kolumnie wyświetlane są nazwa i wartość znacznika.

Aby skonfigurować nowy wyzwalacz, kliknij przycisk *Utwórz wyzwalacz* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

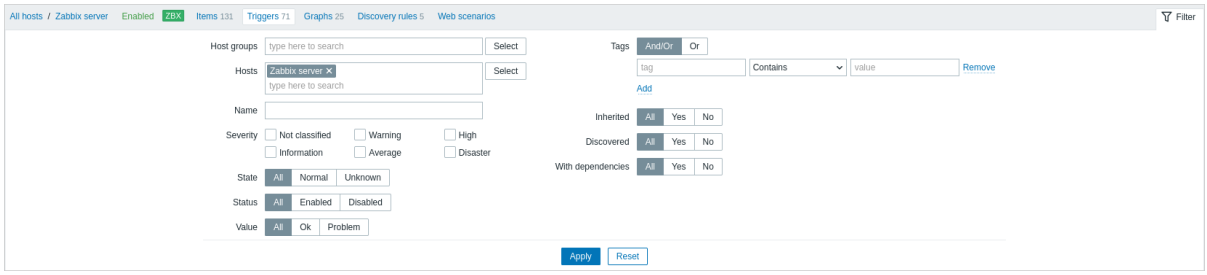
- *Enable* - zmień status wyzwalacza na *Enabled*.
- *Disable* - zmień status wyzwalacza na *Disabled*.
- *Copy* - skopiuj wyzwalacze do innych hostów lub szablonów.
- *Mass update* - zaktualizuj kilka właściwości dla wielu wyzwalaczy jednocześnie.
- *Delete* - usuń wyzwalacze.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi wyzwalaczami, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko wyzwalacze, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane przy nierozwiniętych makrach.

Ikona *Filtr* jest dostępna w prawym górnym rogu. Kliknięcie jej otworzy filtr, w którym możesz określić żądane kryteria filtrowania.



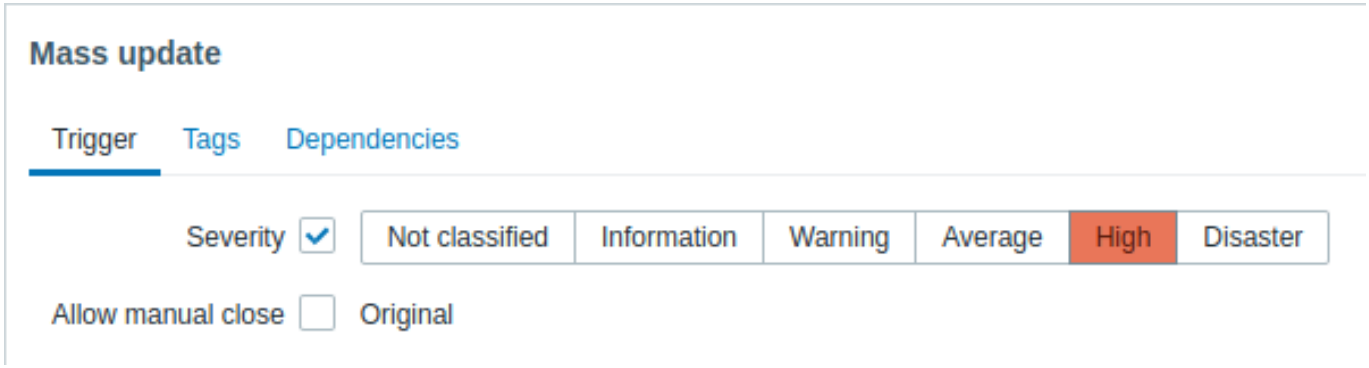
Parameter	Description
Host groups	Filtrowanie według jednej lub wielu grup hostów. Wskazanie nadrzędnej grupy hostów automatycznie zaznacza wszystkie zagnieżdżone grupy hostów. Nie można wybrać grup hostów zawierających wyłącznie szablony.
Hosts	Filtrowanie według jednego lub wielu hostów. Jeśli powyżej wybrano już grupy hostów, wybór hostów jest ograniczony do tych grup.
Name	Filtrowanie według nazwy wyzwalacza.
Severity	Wybierz, aby filtrować według jednej lub kilku wartości ważności wyzwalacza.
State	Filtrowanie według stanu wyrażenia wyzwalacza expression state (Normal/Unknown).
Status	Filtrowanie według statusu aktywacji wyzwalacza (Enabled/Disabled).
Value	Filtrowanie według wartości wyzwalacza.
Tags	Filtrowanie według nazwy i wartości tagu wyzwalacza. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępnych jest kilka operatorów: **Exists** - uwzględnij określone nazwy tagów **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter) **Contains** - uwzględnij określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) **Does not exist** - wyklucz określone nazwy tagów **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter) **Does not contain** - wyklucz określone nazwy tagów, których wartości zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter) Dla warunków dostępne są dwa typy obliczeń: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or **Or** - wystarczy, że spełniony będzie jeden warunek Makra i funkcje makr są obsługiwane zarówno w polu nazwy tagu, jak i wartości tagu.
Inherited	Filtrowanie wyzwalaczy odziedziczonych (lub nieodziedziczonych) z szablonu.
Discovered	Filtrowanie wyzwalaczy wykrytych (lub niewykrytych) przez niskopoziomowe wykrywanie.
With dependencies	Filtrowanie wyzwalaczy z zależnościami (lub bez zależności).

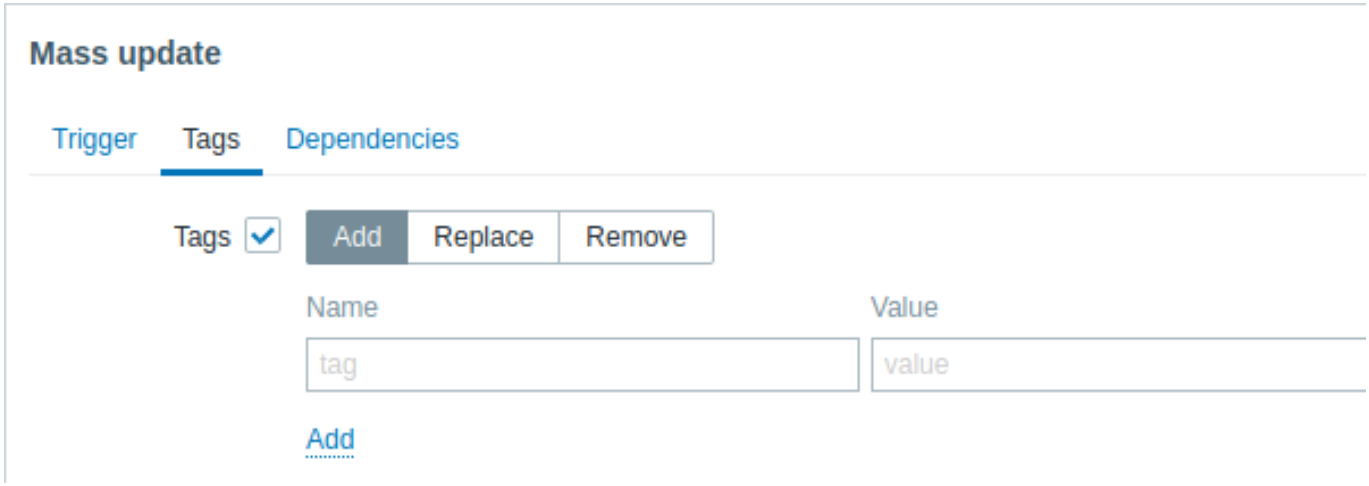
Korzystanie z aktualizacji masowej

Aktualizacja masowa umożliwia jednoczesną zmianę niektórych atrybutów wielu wyzwalaczy, dzięki czemu nie trzeba otwierać każdego wyzwalacza osobno w celu jego edycji.

Aby przeprowadzić aktualizację masową wybranych wyzwalaczy:

- Zaznacz pola wyboru wyzwalaczy, które chcesz zaktualizować na liście
- Kliknij *Aktualizacja masowa* poniżej listy
- Przejdź do karty zawierającej wymagane atrybuty (*Wyzwalacz*, *Znaczniki* lub *Zależności*)
- Zaznacz pola wyboru atrybutów, które chcesz zaktualizować

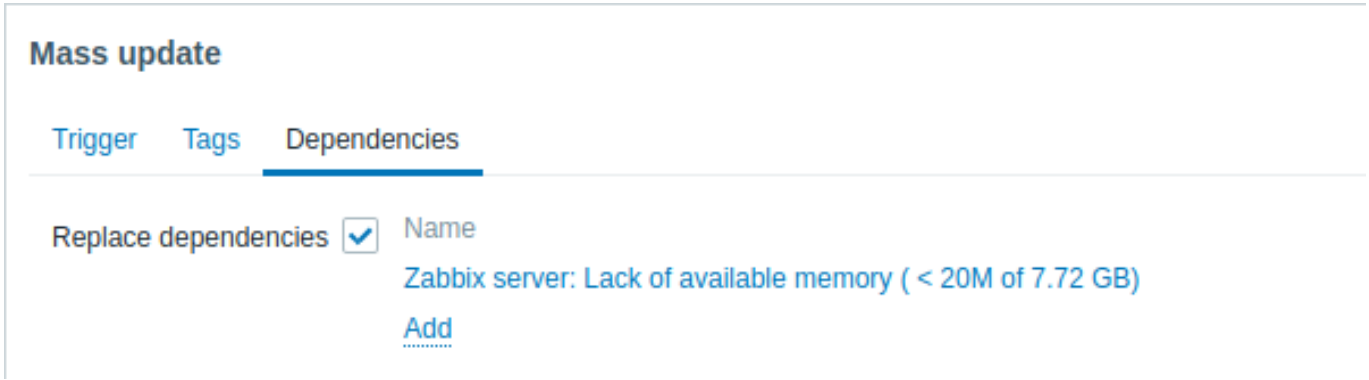




Po wybraniu odpowiedniego przycisku aktualizacji znaczników dostępne są następujące opcje:

- *Dodaj* - dodaje nowe znaczniki do wyzwalaczy;
- *Zastąp* - usuwa wszystkie istniejące znaczniki z wyzwalacza i zastępuje je znacznikami określonymi poniżej;
- *Usuń* - usuwa określone znaczniki z wyzwalaczy.

Należy pamiętać, że znaczniki o tej samej nazwie, ale różnych wartościach, nie są uznawane za „duplikaty” i można je dodać do tego samego wyzwalacza.



Zastąp zależności - usuwa wszystkie istniejące zależności z wyzwalacza i zastępuje je określonymi zależnościami.

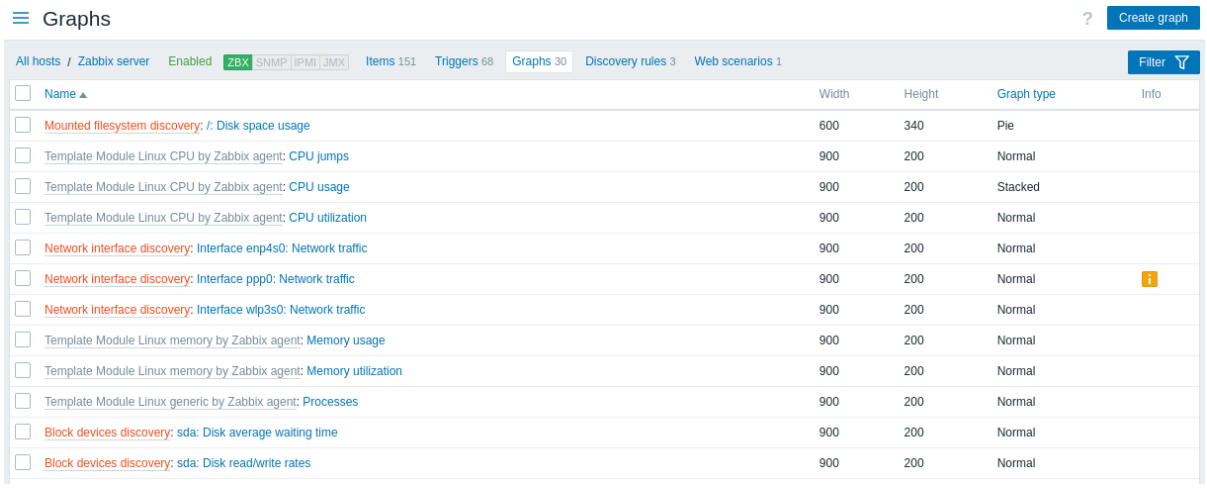
Kliknij *Aktualizuj*, aby zastosować zmiany.

Wykresy

Przegląd

Dostęp do listy niestandardowych wykresów dla hosta można uzyskać, wybierając *Gromadzenie danych* > *Hosty*, a następnie klikając *Wykresy* przy odpowiednim hoście.

Zostanie wyświetlona lista istniejących wykresów.



Wyświetlane dane:

Kolumna	Opis
Host	Host wykresu. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta. Ta kolumna jest wyświetlana tylko wtedy, gdy w filtrze wybrano wiele hostów lub nie wybrano żadnego hosta.
Nazwa	Nazwa niestandardowego wykresu wyświetlana jako niebieski odnośnik do szczegółów wykresu. Kliknięcie odnośnika z nazwą wykresu otwiera formularz konfiguracji wykresu. Jeśli wykres hosta należy do szablonu, przed nazwą wykresu wyświetlana jest nazwa szablonu jako szary odnośnik. Kliknięcie odnośnika szablonu otwiera listę wykresów na poziomie szablonu. Jeśli wykres został utworzony na podstawie prototypu wykresu, jego nazwa jest poprzedzona nazwą reguły wykrywania niskiego poziomu, wyświetlaną na pomarańczowo. Kliknięcie nazwy reguły wykrywania otwiera listę prototypów wykresów.
Szerokość	Wyświetlana jest szerokość wykresu.
Wysokość	Wyświetlana jest wysokość wykresu.
Typ wykresu	Wyświetlany jest typ wykresu: *Zwykły*, *Skumulowany*, *Kołowy* lub *Rozstrzelony*.
Informacje	Jeśli wykres działa prawidłowo, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i". Umieść wskaźnik myszy nad ikoną, aby wyświetlić etykietę z opisem błędu.

Aby skonfigurować nowy wykres, kliknij przycisk *Utwórz wykres* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Copy* - skopiuj wykresy do innych hostów lub szablonów
- *Delete* - usuń wykresy

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi wykresami, a następnie kliknij wymagany przycisk.

Używanie filtra

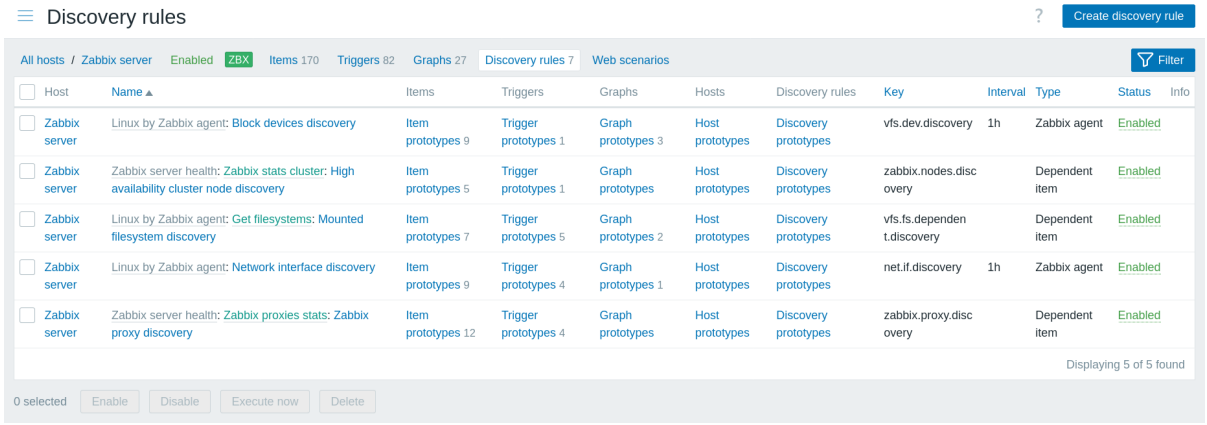
Możesz filtrować wykresy według grupy hostów i hosta. Aby uzyskać lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Reguły wykrywania

Przegląd

Lista reguł wykrywania niskiego poziomu dla hosta jest dostępna w sekcji *Zbieranie danych* > *Hosty* po kliknięciu opcji *Wykrywanie* przy odpowiednim hoście.

Wyświetlana jest lista istniejących reguł wykrywania niskiego poziomu. Można również wyświetlić wszystkie reguły wykrywania niezależnie od hosta lub wszystkie reguły wykrywania należące do określonej grupy hostów, zmieniając ustawienia filtra.



Wyświetlane dane:

Kolumna	Opis
Host	Wyświetlana jest widoczna nazwa hosta. Po kliknięciu nazwy hosta otwierany jest formularz konfiguracji hosta. Jeśli widoczna nazwa hosta nie jest dostępna, wyświetlana jest techniczna nazwa hosta.
Nazwa	Nazwa reguły jest wyświetlana jako niebieski link. Po kliknięciu nazwy reguły otwierany jest formularz konfiguracji reguły wykrywania niskiego poziomu. Jeśli reguła wykrywania należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą reguły jako szary link. Kliknięcie linku szablonu otwiera listę reguł na poziomie szablonu.
Pozycje	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Wyzwalacze	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Wykresy	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosty	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Klucz	Wyświetlany jest klucz pozycji używany do wykrywania.
Interwał	Wyświetlana jest częstotliwość wykonywania wykrywania. *Uwaga*: wykrywanie można również wykonać natychmiast, klikając przycisk *Wykonaj teraz* poniżej listy.
Typ	Wyświetlany jest typ pozycji używany do wykrywania (Zabbix agent, SNMP agent itd.).
Stan	Wyświetlany jest stan reguły wykrywania: *Włączona*, *Wyłączona* lub *Nieobsługiwana*. Klikając stan, można go zmienić: z Włączona na Wyłączona (i odwrotnie) oraz z Nieobsługiwana na Wyłączona (i odwrotnie).
Informacje	Jeśli wszystko działa prawidłowo, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i". Umieść wskaźnik myszy nad ikoną, aby wyświetlić etykietę z opisem błędu.

Aby skonfigurować nową regułę wykrywania niskiego poziomu, kliknij przycisk *Utwórz regułę wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

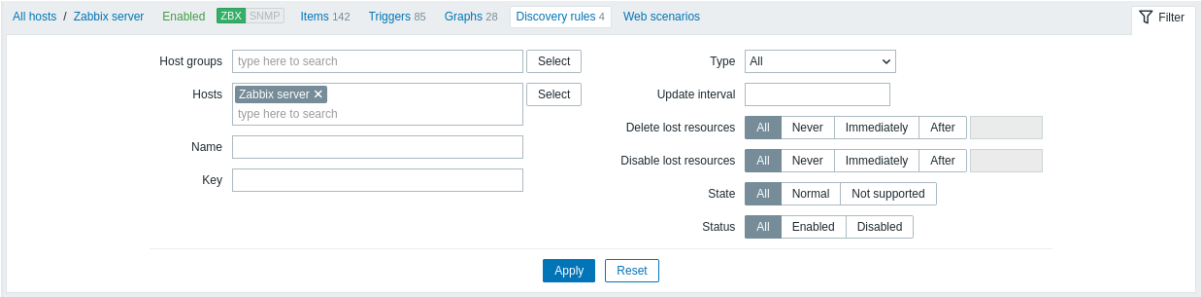
- *Włącz* - zmień status reguły wykrywania niskiego poziomu na *Włączony*.
- *Wyłącz* - zmień status reguły wykrywania niskiego poziomu na *Wyłączony*.
- *Wykonaj teraz* - natychmiast wykonaj wykrywanie na podstawie reguł wykrywania. Zobacz więcej szczegółów. Pamiętaj, że podczas natychmiastowego wykonywania wykrywania pamięć podręczna konfiguracji nie jest aktualizowana, więc wynik nie będzie odzwierciedlał bardzo niedawnych zmian w konfiguracji reguły wykrywania.
- *Usuń* - usuń reguły wykrywania niskiego poziomu.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi regułami wykrywania, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły wykrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Łącze *Filter* jest dostępne nad listą reguł wykrywania. Po jego kliknięciu pojawia się filtr, w którym można filtrować reguły wykrywania według grup hostów, hosta, nazwy, klucza pozycji, typu pozycji i innych parametrów.

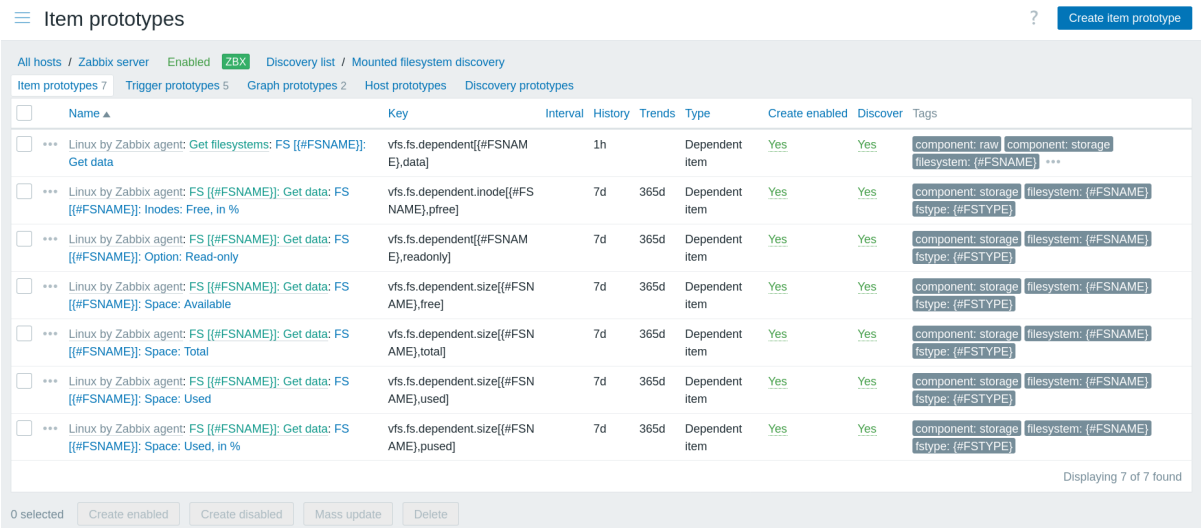


Parameter	Description
Host groups	Filtrowanie według jednej lub wielu grup hostów. Określenie nadrzędnej grupy hostów automatycznie zaznacza wszystkie zagnieżdżone grupy hostów.
Hosts	Filtrowanie według jednego lub wielu hostów.
Name	Filtrowanie według nazwy reguły wykrywania.
Key	Filtrowanie według klucza pozycji wykrywania.
Type	Filtrowanie według typu pozycji wykrywania.
Update interval	Filtrowanie według interwału aktualizacji. Niedostępne dla elementów Zabbix trapper i zależnych.
Delete lost resources	Filtrowanie według okresu *Delete lost resources*.
Disable lost resources	Filtrowanie według okresu *Disable lost resources*.
SNMP OID	Filtrowanie według SNMP OID. Dostępne tylko wtedy, gdy jako typ wybrano *SNMP agent*.
State	Filtrowanie według stanu reguły wykrywania (All/Normal/Not supported).
Status	Filtrowanie według statusu reguły wykrywania (All/Enabled/Disabled).

Prototypy pozycji

Przegląd

W tej sekcji wyświetlane są prototypy pozycji reguły wykrywania niskopoziomowego na hoście. Prototypy pozycji stanowią podstawę rzeczywistych pozycji hosta, które są tworzone podczas wykrywania niskopoziomowego.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu pozycji wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu pozycji. Jeśli prototyp pozycji należy do szablonu, nazwa szablonu jest wyświetlana przed nazwą reguły jako szary link. Kliknięcie linku szablonu otwiera listę prototypów pozycji na poziomie szablonu. Jeśli prototyp pozycji jest typu „Zależna pozycja”, nazwa pozycji nadrzędnej jest wyświetlana przed nazwą prototypu pozycji na zielono.
Klucz	Wyświetlany jest klucz prototypu pozycji.
Interwał	Wyświetlana jest częstotliwość sprawdzania.
Historia	Wyświetlana jest liczba dni przechowywania historii danych pozycji.
Trendy	Wyświetlana jest liczba dni przechowywania historii trendów pozycji.
Typ	Wyświetlany jest typ prototypu pozycji (agent Zabbix, agent SNMP, proste sprawdzenie itd.).
Tworzenie włączone	Tworzenie pozycji na podstawie tego prototypu jako: **Tak** - włączone **Nie** - wyłączone. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Wykrywaj	Wykrywanie pozycji na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Znaczniki	Wyświetlane są znaczniki prototypu pozycji.

Aby skonfigurować nowy prototyp pozycji, kliknij przycisk *Utwórz prototyp pozycji* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

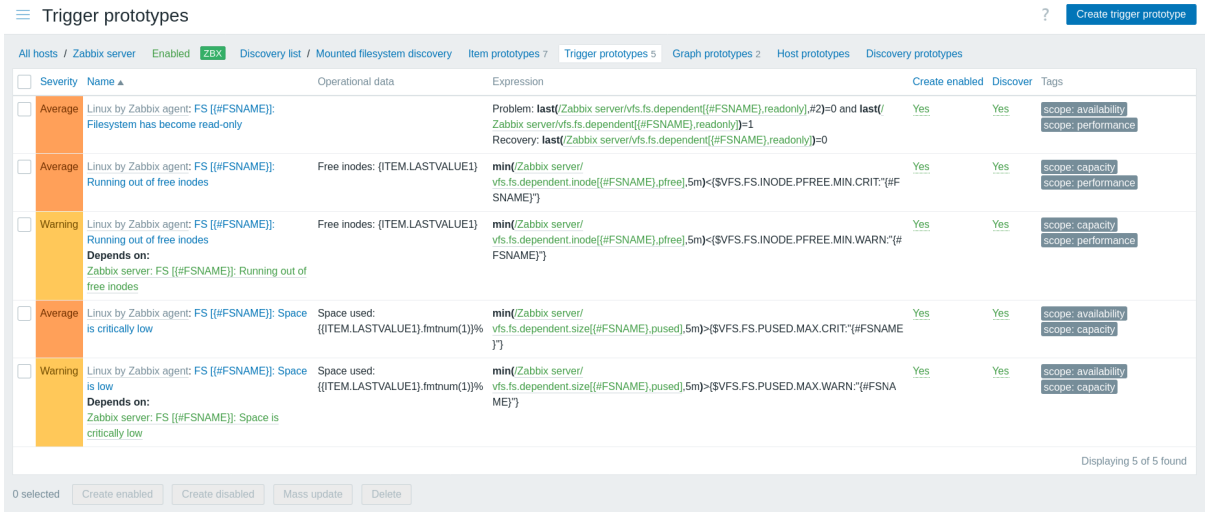
- *Utwórz jako włączone* - utwórz te pozycje jako *Włączone*
- *Utwórz jako wyłączone* - utwórz te pozycje jako *Wyłączone*
- *Masowa aktualizacja* - masowo zaktualizuj te prototypy pozycji
- *Usuń* - usuń te prototypy pozycji

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami pozycji, a następnie kliknij wymagany przycisk.

Prototypy wyzwalaczy

Przegląd

W tej sekcji wyświetlane są prototypy wyzwalaczy reguły wykrywania niskiego poziomu na hoście. Prototypy wyzwalaczy stanowią podstawę rzeczywistych wyzwalaczy hosta, które są tworzone podczas wykrywania niskiego poziomu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wyzwalacza wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wyzwalacza. Jeśli prototyp wyzwalacza należy do połączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wyzwalacza jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wyzwalaczy na poziomie połączonego szablonu.
Dane operacyjne	Wyświetlany jest format danych operacyjnych wyzwalacza, zawierający dowolne ciągi znaków i makra, które zostaną dynamicznie rozwinięte w sekcji *Monitorowanie* > *Problemy*.
Tworzenie włączone	Utwórz wyzwalacz na podstawie tego prototypu jako: **Tak** - włączony **Nie** - wyłączony. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Wykrywaj	Wykrywaj wyzwalacz na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Znaczniki	Wyświetlane są znaczniki prototypu wyzwalacza.

Aby skonfigurować nowy prototyp wyzwalacza, kliknij przycisk *Utwórz prototyp wyzwalacza* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

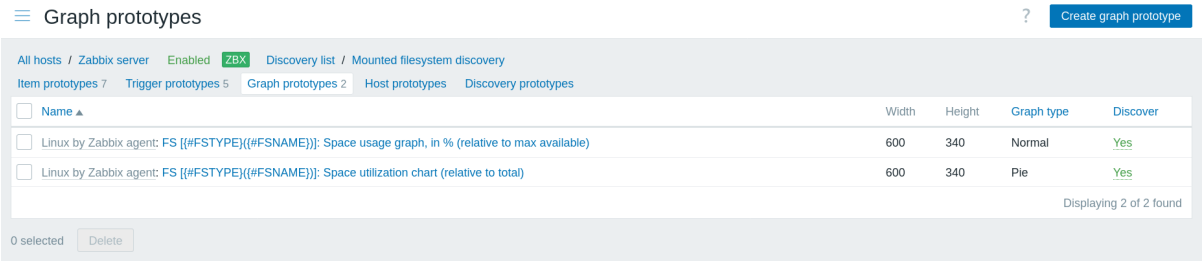
- *Utwórz włączone* - utwórz te wyzwalacze jako *Włączone*
- *Utwórz wyłączone* - utwórz te wyzwalacze jako *Wyłączone*
- *Masowa aktualizacja* - wykonaj masową aktualizację tych prototypów wyzwalaczy
- *Usuń* - usuń te prototypy wyzwalaczy

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami wyzwalaczy, a następnie kliknij wymagany przycisk.

Prototypy wykresów

Przegląd

W tej sekcji wyświetlane są prototypy wykresów reguły wykrywania niskopoziomowego na hoście. Prototypy wykresów stanowią podstawę rzeczywistych wykresów hosta, które są tworzone podczas wykrywania niskopoziomowego.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu wykresu wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykresu. Jeśli prototyp wykresu należy do połączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą wykresu jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykresów na poziomie połączonego szablonu.
Szerokość	Wyświetlana jest szerokość prototypu wykresu.
Wysokość	Wyświetlana jest wysokość prototypu wykresu.
Typ	Wyświetlany jest typ prototypu wykresu: *Normalny*, *Skumulowany*, *Kołowy* lub *Rozstrzelony*.
Wykrywaj	Wykrywanie wykresu na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Klikając te wartości, można przełączać się między „Tak” i „Nie”.

Aby skonfigurować nowy prototyp wykresu, kliknij przycisk *Utwórz prototyp wykresu* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

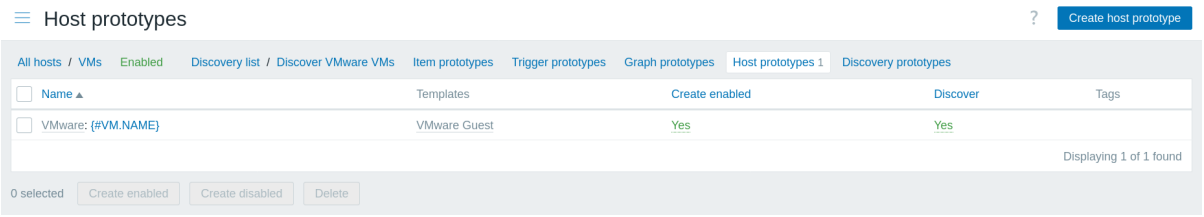
- *Usuń* - usuń te prototypy wykresów

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami wykresów, a następnie kliknij wymagany przycisk.

Prototypy hostów

Przegląd

W tej sekcji wyświetlane są prototypy hostów reguły wykrywania niskopoziomowego na hoście. Prototypy hostów stanowią podstawę rzeczywistych hostów, które są tworzone podczas wykrywania niskopoziomowego.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa prototypu hosta wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu hosta. Jeśli prototyp hosta należy do połączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą hosta jako szary link. Kliknięcie linku szablonu otwiera listę prototypów hostów na poziomie połączonego szablonu.
Szablony	Wyświetlane są szablony prototypu hosta.
Tworzenie włączone	Utwórz host na podstawie tego prototypu jako: **Tak** - włączone **Nie** - wyłączone. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Wykrywanie	Wykryj host na podstawie tego prototypu: **Tak** - wykrywaj **Nie** - nie wykrywaj. Możesz przełączać się między wartościami „Tak” i „Nie”, klikając je.
Znaczniki	Wyświetlane są znaczniki prototypu hosta.

Aby skonfigurować nowy prototyp hosta, kliknij przycisk *Utwórz prototyp hosta* w prawym górnym rogu.

Opcje masowej edycji

Przyciski pod listą oferują kilka opcji masowej edycji:

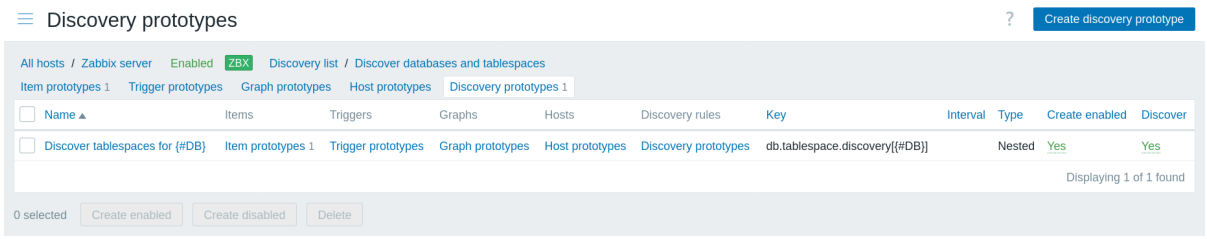
- *Utwórz włączone* - utwórz te hosty jako *Włączone*
- *Utwórz wyłączone* - utwórz te hosty jako *Wyłączone*
- *Usuń* - usuń te prototypy hostów

Aby użyć tych opcji, zaznacz pola wyboru przy odpowiednich prototypach hostów, a następnie kliknij wymagany przycisk.

Protokoły wykrywania

Przegląd

W tej sekcji wyświetlane są prototypy wykrywania reguły wykrywania niskiego poziomu na hoście. Prototypy wykrywania są zagnieżdżonymi regułami wykrywania nadrzędnej reguły wykrywania.



Wyświetlane dane:

Column	Description
Name	Nazwa prototypu wykrywania wyświetlana jako niebieski link. Kliknięcie nazwy otwiera formularz konfiguracji prototypu wykrywania. Jeśli prototyp wykrywania należy do połączonego szablonu, nazwa szablonu jest wyświetlana przed nazwą hosta jako szary link. Kliknięcie linku szablonu otwiera listę prototypów wykrywania na poziomie połączonego szablonu.
Items	Wyświetlany jest link do listy prototypów pozycji. Liczba istniejących prototypów pozycji jest wyświetlana na szaro.
Triggers	Wyświetlany jest link do listy prototypów wyzwalaczy. Liczba istniejących prototypów wyzwalaczy jest wyświetlana na szaro.
Graphs	Wyświetlany jest link do listy prototypów wykresów. Liczba istniejących prototypów wykresów jest wyświetlana na szaro.
Hosts	Wyświetlany jest link do listy prototypów hostów. Liczba istniejących prototypów hostów jest wyświetlana na szaro.
Discovery rules	Wyświetlany jest link do listy prototypów reguł wykrywania. Liczba istniejących prototypów reguł wykrywania jest wyświetlana na szaro.
Key	Wyświetlany jest klucz pozycji używany do wykrywania.
Interval	Wyświetlana jest częstotliwość wykonywania wykrywania. *Uwaga*: wykrywanie można również wykonać natychmiast, naciskając przycisk *Execute now* poniżej listy.
Type	Wyświetlany jest typ pozycji używany do wykrywania (agent Zabbix, agent SNMP itd.).
Create enabled	Wyświetlany jest status tworzenia prototypu wykrywania - tworzenie włączone (*Yes*) lub tworzenie wyłączone (*No*). Klikając status, można go zmienić z Yes na No (i odwrotnie).
Discover	Wyświetlany jest status wykrywania prototypu wykrywania - wykrywaj (*Yes*) lub nie wykrywaj (*No*). Klikając status, można go zmienić z Yes na No (i odwrotnie).

Aby skonfigurować nowy prototyp wykrywania, kliknij przycisk *Create discovery prototype* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Utwórz włączone* - utwórz te prototypy wykrywania jako *Włączone*
- *Utwórz wyłączone* - utwórz te prototypy wykrywania jako *Wyłączone*
- *Usuń* - usuń te prototypy wykrywania

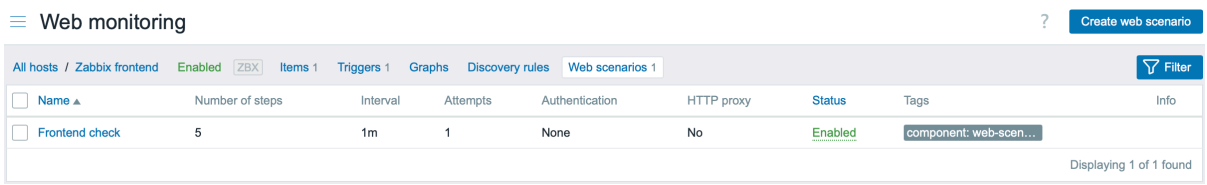
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi prototypami wykrywania, a następnie kliknij wymagany przycisk.

Scenariusze webowe

Przegląd

Listę scenariuszy internetowych dla hosta można otworzyć, wybierając *Gromadzenie danych* > *Hosty*, a następnie klikając *Web* przy odpowiednim hoście.

Zostanie wyświetlona lista istniejących scenariuszy internetowych.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa scenariusza internetowego. Kliknięcie nazwy scenariusza internetowego otwiera formularz konfiguracji scenariusza internetowego. Jeśli scenariusz internetowy hosta należy do szablonu, przed nazwą scenariusza internetowego wyświetlana jest nazwa szablonu jako szary link. Kliknięcie linku szablonu otworzy listę scenariuszy internetowych na poziomie szablonu.
Liczba kroków	Liczba kroków zawartych w scenariuszu.
Interwał aktualizacji	Częstotliwość wykonywania scenariusza.
Próby	Liczba prób wykonania kroków scenariusza internetowego.
Uwierzytelnianie	Wyświetlana jest metoda uwierzytelniania: Basic, NTLM, Kerberos, Digest lub None.
Proxy HTTP	Wyświetlany jest proxy HTTP lub „No”, jeśli nie jest używany.
Stan	Wyświetlany jest stan scenariusza internetowego: *Enabled* lub *Disabled*. Klikając stan, można go zmienić.
Tagi	Wyświetlane są tagi scenariusza internetowego. Można wyświetlić maksymalnie trzy tagi (pary nazwa:wartość). Jeśli tagów jest więcej, wyświetlany jest link „...”, który umożliwia wyświetlenie wszystkich tagów po najechaniu kursorem myszy.
Informacje	Jeśli wszystko działa prawidłowo, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów wyświetlana jest kwadratowa ikona z literą „i”. Umieść kursor nad ikoną, aby wyświetlić etykietę z opisem błędu.

Aby skonfigurować nowy scenariusz internetowy, kliknij przycisk *Utwórz scenariusz internetowy* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Włącz* - zmień status scenariusza na *Włączony*
- *Wyłącz* - zmień status scenariusza na *Wyłączony*
- *Wyczyść historię i trendy* - wyczyść dane historii i trendów dla scenariuszy
- *Usuń* - usuń scenariusze web

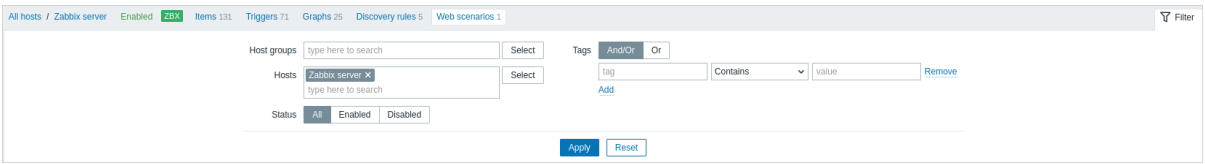
Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi scenariuszami web, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko scenariusze, które Cię interesują.

Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Link *Filter* jest dostępny nad listą scenariuszy webowych. Kliknięcie go otworzy filtr, w którym możesz określić żądane kryteria filtrowania.



Parameter	Description
Host groups	Filtrowanie według jednej lub wielu grup hostów. Wskazanie nadrzędnej grupy hostów automatycznie wybiera wszystkie zagnieżdżone grupy hostów.
Hosts	Filtrowanie według jednego lub wielu hostów. Jeśli grupy hostów są już wybrane powyżej, wybór hostów jest ograniczony do tych grup.
Status	Filtrowanie według statusu scenariusza webowego.

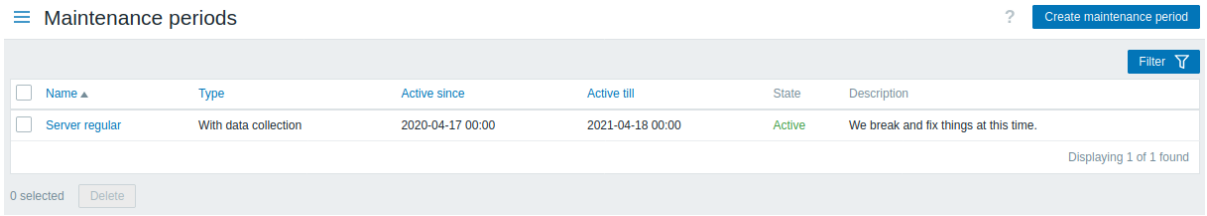
Parameter	Description
Tags	Filtrowanie według nazwy i wartości tagu scenariusza webowego. Możliwe jest zarówno uwzględnianie, jak i wykluczanie określonych tagów oraz wartości tagów. Można ustawić kilka warunków. Dopasowanie nazwy tagu jest zawsze rozróżniane wielkością liter. Dla każdego warunku dostępne są następujące operatory: **Exists** - uwzględnij określone nazwy tagów; **Equals** - uwzględnij określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Contains** - uwzględnij określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter); **Does not exist** - wyklucz określone nazwy tagów; **Does not equal** - wyklucz określone nazwy tagów i wartości (z rozróżnianiem wielkości liter); **Does not contain** - wyklucz określone nazwy tagów, których wartości tagów zawierają wprowadzony ciąg znaków (dopasowanie podciągu, bez rozróżniania wielkości liter). Dostępne są dwa typy obliczania warunków: **And/Or** - wszystkie warunki muszą być spełnione, warunki o tej samej nazwie tagu będą grupowane według warunku Or; **Or** - wystarczy, że spełniony będzie jeden warunek.

Konserwacja

Przegląd

W sekcji *Zbieranie danych* > *Okresy konserwacji* użytkownicy mogą konfigurować i zarządzać okresami konserwacji dla hostów.

Wyświetlana jest lista istniejących okresów konserwacji wraz ze szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa okresu konserwacji. Kliknięcie nazwy okresu konserwacji otwiera formularz konfiguracji okresu konserwacji.
Typ	Wyświetlany jest typ konserwacji: *Ze zbieraniem danych* lub *Bez zbierania danych*
Aktywny od	Data i godzina, od której wykonywanie okresu konserwacji staje się aktywne. Uwaga: ten czas nie aktywuje okresu konserwacji; okresy konserwacji należy ustawić osobno.
Aktywny do	Data i godzina, do której wykonywanie okresu konserwacji pozostaje aktywne.
Stan	Stan okresu konserwacji: **Nadchodzący** - wkrótce stanie się aktywny **Aktywny** - jest aktywny **Wygasły** - nie jest już aktywny
Opis	Wyświetlany jest opis okresu konserwacji.

Aby skonfigurować nowy okres konserwacji, kliknij przycisk *Utwórz okres konserwacji* w prawym górnym rogu.

Opcje masowej edycji

Przycisk poniżej listy oferuje jedną opcję masowej edycji:

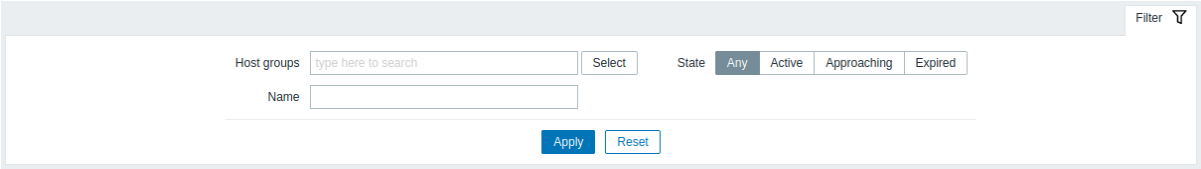
- *Usuń* - usuń okresy konserwacji

Aby użyć tej opcji, zaznacz pola wyboru przed odpowiednimi okresami konserwacji i kliknij *Usuń*.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te okresy konserwacji, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Łącze *Filter* jest dostępne nad listą okresów konserwacji. Po kliknięciu pojawi się filtr, za pomocą którego można filtrować okresy konserwacji według grupy hostów, nazwy i stanu.



Obliczanie kolejek podczas konserwacji

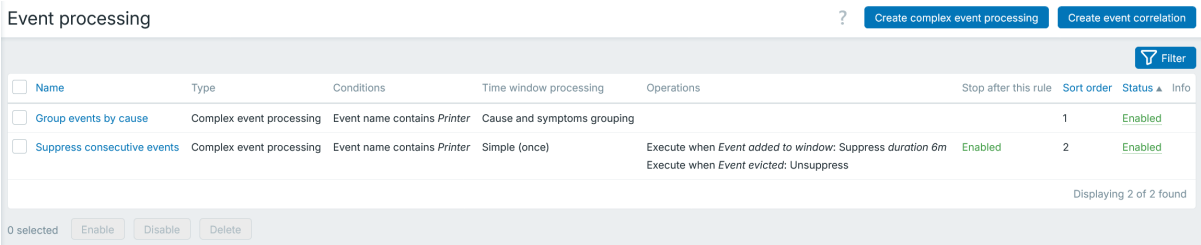
Attention:
Proxy Zabbix nie jest świadomy okresów konserwacji; szczegóły można znaleźć w sekcji Obliczanie kolejek podczas konserwacji.

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:

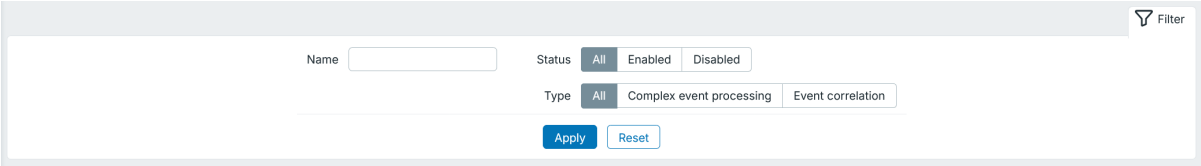
- *Enable* - change the rule status to *Enabled*
- *Disable* - change the rule status to *Disabled*
- *Delete* - delete the rules

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.

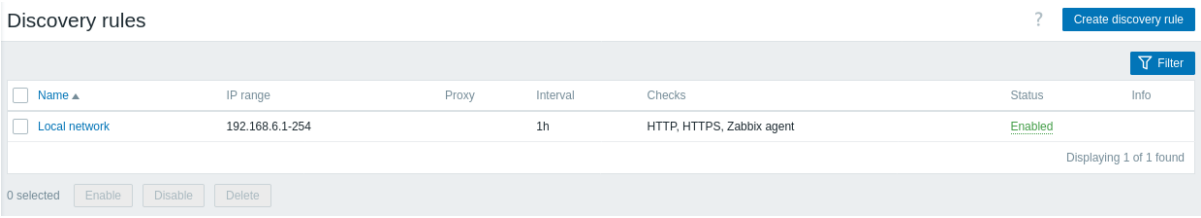


Odkrywanie

Przegląd

W sekcji *Zbieranie danych* > *Wykrywanie* użytkownicy mogą konfigurować i utrzymywać reguły wykrywania.

Wyświetlana jest lista istniejących reguł wykrywania wraz ze szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa reguły wykrywania. Kliknięcie nazwy reguły wykrywania otwiera formularz konfiguracji reguły wykrywania.
Zakres adresów IP	Wyświetlany jest zakres adresów IP używany do skanowania sieci.
Proxy	Wyświetlana jest nazwa proxy, jeśli wykrywanie jest wykonywane przez proxy.
Interwał	Wyświetlana jest częstotliwość wykonywania wykrywania.
Sprawdzenia	Wyświetlane są typy sprawdzeń używanych podczas wykrywania.
Stan	Wyświetlany jest stan reguły wykrywania - *Włączona* lub *Wyłączona*. Klikając stan, można go zmienić.
Informacje	Jeśli wszystko działa prawidłowo, w tej kolumnie nic nie jest wyświetlane. W przypadku błędów wyświetlana jest czerwona ikona informacji z literą „i”. Umieść wskaźnik myszy nad ikoną, aby wyświetlić etykietę narzędzia z opisem błędu.

Aby skonfigurować nową regułę wykrywania, kliknij przycisk *Utwórz regułę wykrywania* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

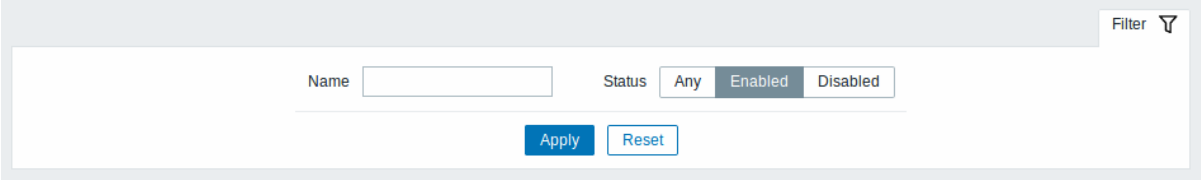
- *Włącz* - zmienia status reguły wykrywania na *Włączona*
- *Wyłącz* - zmienia status reguły wykrywania na *Wyłączona*
- *Usuń* - usuwa reguły wykrywania

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi regułami wykrywania, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te reguły wykrywania, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Łącze *Filter* jest dostępne nad listą reguł wykrywania. Po jego kliknięciu staje się dostępny filtr, za pomocą którego można filtrować reguły wykrywania według nazwy i statusu.



Alerty

Przegląd

To menu zawiera sekcje związane z konfiguracją alertów w Zabbix.

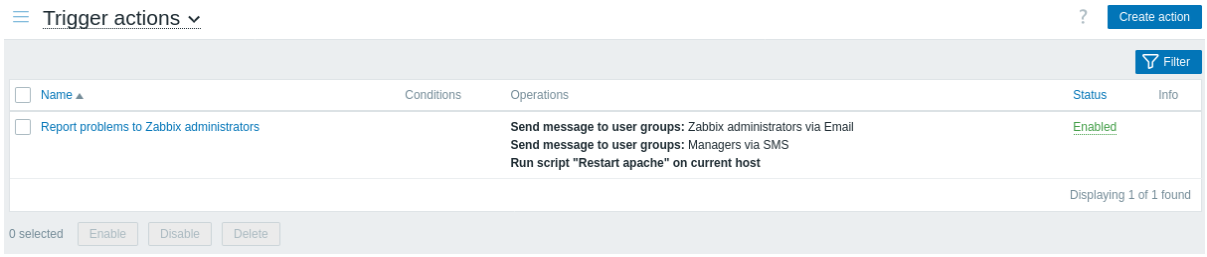
Akcje

Przegląd

W sekcji *Alerty* > *Działania* użytkownicy mogą konfigurować i zarządzać działaniami.

Wyświetlane są działania przypisane do wybranego źródła zdarzeń (wyzwalacze, usługi, wykrywanie, automatyczna rejestracja, działania wewnętrzne).

Aby przełączyć się na inne źródło zdarzeń, kliknij *Działania* w sekcji menu *Alerty*. Można również przełączać się między źródłami za pomocą listy rozwijanej tytułu w lewym górnym rogu.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa działania. Kliknięcie nazwy działania otwiera formularz konfiguracji działania.
Warunki	Wyświetlane są warunki działania.
Operacje	Wyświetlane są operacje działania. Lista operacji zawiera również typ nośnika (e-mail, SMS lub skrypt) używanego do wysyłania powiadomień, a także imię i nazwisko (w nawiasach po nazwie użytkownika) odbiorcy powiadomienia. Operacja działania może być powiadomieniem lub zdalnym poleceniem, zależnie od wybranego typu operacji.
Stan	Wyświetlany jest stan działania - *Włączone* lub *Wyłączone*. Klikając stan, można go zmienić. Więcej informacji o tym, co się dzieje, gdy działanie zostanie wyłączone podczas trwającej eskalacji, można znaleźć w sekcji Eskalacje.
Informacje	Jeśli wszystko działa prawidłowo, w tej kolumnie nie jest wyświetlana żadna ikona. W przypadku błędów, takich jak brak operacji lub warunków działania po aktualizacji/wykrywaniu niskopoziomowym, wyświetlana jest ikona ostrzeżenia. Umieść wskaźnik myszy nad ikoną, aby wyświetlić etykietę z opisem błędu.

Aby skonfigurować nowe działanie, kliknij przycisk *Utwórz działanie* w prawym górnym rogu.

W przypadku użytkowników bez uprawnień administratora głównego działania są wyświetlane zgodnie z ustawieniami uprawnień. Oznacza to, że w niektórych przypadkach użytkownik bez uprawnień administratora głównego nie może wyświetlić pełnej listy działań z powodu określonych ograniczeń uprawnień. Działanie jest wyświetlane użytkownikowi bez uprawnień administratora głównego, jeśli spełnione są następujące warunki:

- Użytkownik ma dostęp do odczytu i zapisu grup hostów, hostów, szablonów i wyzwalaczy w warunkach działania
- Użytkownik ma dostęp do odczytu i zapisu grup hostów, hostów i szablonów w operacjach działania, operacjach odzyskiwania i operacjach aktualizacji
- Użytkownik ma dostęp do odczytu grup użytkowników i użytkowników w operacjach działania, operacjach odzyskiwania i operacjach aktualizacji

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

- *Włącz* - zmienia status akcji na *Włączona*
- *Wyłącz* - zmienia status akcji na *Wyłączona*

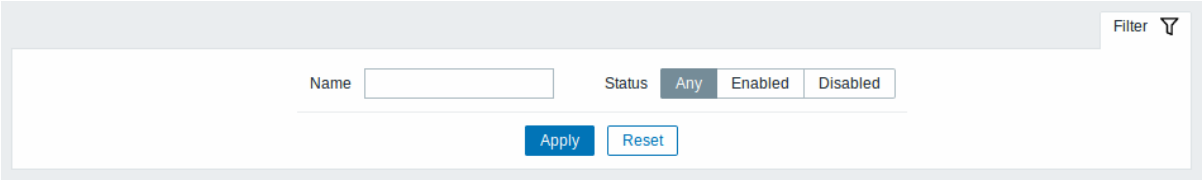
- *Usuń* - usuwa akcje

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi akcjami, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te akcje, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Link *Filtr* jest dostępny nad listą akcji. Po kliknięciu udostępniony zostanie filtr, w którym można filtrować akcje według nazwy i statusu.



Typy mediów

Przegląd

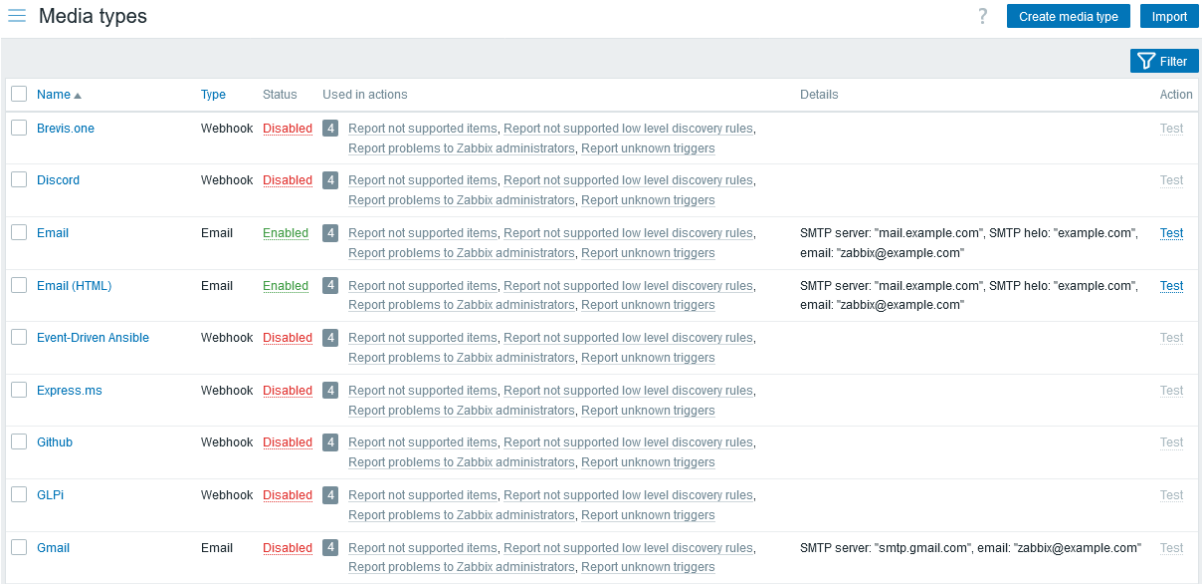
W sekcji *Alerty* > *Typy mediów* użytkownicy mogą konfigurować informacje o typach mediów i zarządzać nimi.

To menu jest dostępne wyłącznie dla użytkowników typu SuperAdmin.

Informacje o typie mediów zawierają ogólne instrukcje dotyczące używania medium jako kanału dostarczania powiadomień. Szczegółowe informacje, takie jak poszczególne adresy e-mail, na które należy wysłać powiadomienie, są przechowywane przy poszczególnych użytkownikach.

Wyświetlana jest lista istniejących typów mediów wraz z ich szczegółami.

Note:
Jeśli metody dostarczania (*Email*, *SMS*, *Webhook*, *Script* lub *Push*) zostały określone we fladze funkcji `$ZBX_FEATURE_FLAGS['media_type_denylist']` w pliku konfiguracyjnym frontendu, odpowiadające im wstępnie skonfigurowane typy mediów są wyłączone w tej sekcji.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa typu mediów. Kliknięcie nazwy otwiera formularz konfiguracji typu mediów.
Typ	Wyświetlany jest typ medium (e-mail, SMS itd.).
Stan	Wyświetlany jest stan typu mediów - *Włączony* lub *Wyłączony*. Kliknięcie stanu umożliwia jego zmianę.

Kolumna	Opis
Używany w działaniach	Wyświetlane są działania, w których używany jest typ mediów, poprzedzone łączną liczbą tych działań. Kliknięcie nazwy działania otwiera formularz konfiguracji działania. Jeśli użytkownik nie ma uprawnień do działania, nazwy nie można kliknąć.
Szczegóły	Wyświetlane są szczegółowe informacje o typie mediów.
Działania	Dostępne jest następujące działanie: **Test** - kliknij, aby otworzyć formularz testowania, w którym można wprowadzić parametry typu mediów (np. adres odbiorcy oraz temat i treść testowej wiadomości) i wysłać testową wiadomość w celu sprawdzenia, czy skonfigurowany typ mediów działa. Zobacz także: Testowanie typu mediów.

Aby skonfigurować nowy typ mediów, kliknij przycisk *Utwórz typ mediów* w prawym górnym rogu.

Aby zaimportować typ mediów, kliknij przycisk *Importuj* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

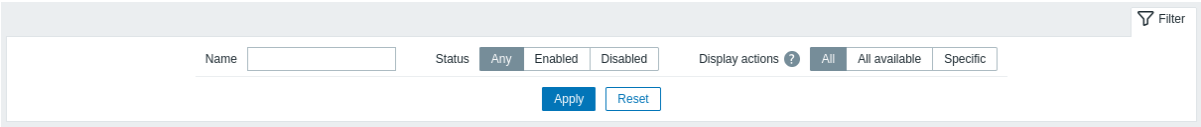
- *Włącz* - zmienia status typu mediów na *Włączony*
- *Wyłącz* - zmienia status typu mediów na *Wyłączony*
- *Eksportuj* - eksportuje typy mediów do pliku YAML, XML lub JSON
- *Usuń* - usuwa typy mediów

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi typami mediów, a następnie kliknij wymagany przycisk.

Używanie filtra

Możesz użyć filtra, aby wyświetlić tylko te typy mediów, które Cię interesują. Dla lepszej wydajności wyszukiwania dane są przeszukiwane z nierozwiązanymi makrami.

Link *Filter* jest dostępny nad listą typów mediów. Po jego kliknięciu staje się dostępny filtr, za pomocą którego można filtrować typy mediów według nazwy i statusu. Dodatkowo możesz użyć filtra, aby wyświetlać akcje w kolumnie *Used in actions* na podstawie zakresu użycia ich typu mediów (określonego przez parametr *Send to media type* w szczegółach operacji akcji).



Skrypty

Przegląd

W sekcji *Alerts* > *Scripts* można konfigurować i zarządzać zdefiniowanymi przez użytkownika skryptami globalnymi.

To menu jest dostępne wyłącznie dla użytkowników typu SuperAdmin.

Skrypty globalne, w zależności od skonfigurowanego zakresu, a także uprawnień użytkownika, są dostępne do wykonania:

- Z poziomu menu hosta w różnych miejscach frontendu (*Dashboard*, *Problems*, *Latest data*, *Maps* itd.).
- Z poziomu menu zdarzenia.
- Mogą być uruchamiane jako operacja akcji.

Skrypty są wykonywane wyłącznie przez Zabbix agent, Zabbix serwer (proxy) lub Zabbix serwer. Zobacz także Wykonywanie poleceń.

Polecenia zdalne są domyślnie wyłączone zarówno w Zabbix agent, jak i w Zabbix proxy. Można je włączyć w następujący sposób:

- W przypadku poleceń zdalnych wykonywanych przez Zabbix agent dodaj parametr `AllowKey=system.run[<command>,*]` lub `AllowKeyRegexp=^system\.run\[<command>,.*\]$` dla każdego dozwolonego polecenia w konfiguracji agenta, gdzie `*` odpowiada dowolnemu trybowi wykonywania (`wait` lub `nowait`).
- W przypadku poleceń zdalnych wykonywanych przez Zabbix proxy ustaw parametr `EnableRemoteCommands` na `1` w konfiguracji proxy. Należy pamiętać, że nie trzeba włączać poleceń zdalnych w Zabbix proxy, gdy są one wykonywane przez Zabbix agent na hoście monitorowanym przez proxy.

Attention:
Nie używaj `AllowKey=system.run[*]` ani `AllowKeyRegexp=^system\.run\[.*\]$`, chyba że jest to konieczne, ponieważ umożliwia to wykonywanie dowolnych poleceń na hoście. Ze względów bezpieczeństwa zaleca się zezwalanie wyłącznie na dokładnie określone polecenia, natomiast `system.run[*]` jest domyślnie blokowane (`DenyKey=system.run[*]`).

Wykonywanie skryptów globalnych na Zabbix serwer można wyłączyć, ustawiając `EnableGlobalScripts=0` w konfiguracji serwera. W przypadku nowych instalacji rozpoczynających się od Zabbix 7.0 wykonywanie skryptów globalnych na Zabbix serwer jest domyślnie wyłączone.

Wyświetlana jest lista istniejących skryptów wraz ze szczegółami.



Wyświetlane dane:

Kolumna	Opis
Name	Nazwa skryptu. Kliknięcie nazwy skryptu otwiera formularz konfiguracji skryptu.
Scope	Zakres skryptu - operacja akcji, ręczna akcja hosta lub ręczna akcja zdarzenia. To ustawienie określa, gdzie skrypt jest dostępny - w poleceniach zdalnych operacji akcji, z poziomu menu hosta lub z poziomu menu zdarzenia.
Used in actions	Wyświetlane są wszystkie akcje, w których używany jest skrypt, poprzedzone łączną liczbą tych akcji. Kliknięcie nazwy akcji otwiera formularz konfiguracji akcji. Jeśli użytkownik nie ma uprawnień do akcji, nazwa nie jest klikalna.
Type	Wyświetlany jest typ skryptu - polecenie typu *URL*, *Webhook*, *Script*, *SSH*, *Telnet* lub *IPMI*.
Execute on	Wyświetlana jest informacja, czy skrypt będzie wykonywany przez Zabbix agent, Zabbix proxy lub serwer, czy tylko przez Zabbix serwer.
Commands	Wyświetlane są wszystkie polecenia, które mają zostać wykonane w ramach skryptu. W przypadku webhooków nic nie jest tutaj wyświetlane.
User group	Wyświetlana jest grupa użytkowników, dla której skrypt jest dostępny (lub *All* w przypadku wszystkich grup użytkowników).
Host group	Wyświetlana jest grupa hostów, dla której skrypt jest dostępny (lub *All* w przypadku wszystkich grup hostów).
Host access	Wyświetlany jest poziom uprawnień dla grupy hostów - *Read* lub *Write*. Dostęp do wykonywania skryptu będą mieli tylko użytkownicy z wymaganym poziomem uprawnień.

Aby skonfigurować nowy skrypt, kliknij przycisk *Create script* w prawym górnym rogu.

Opcje masowej edycji

Przycisk poniżej listy oferuje jedną opcję masowej edycji:

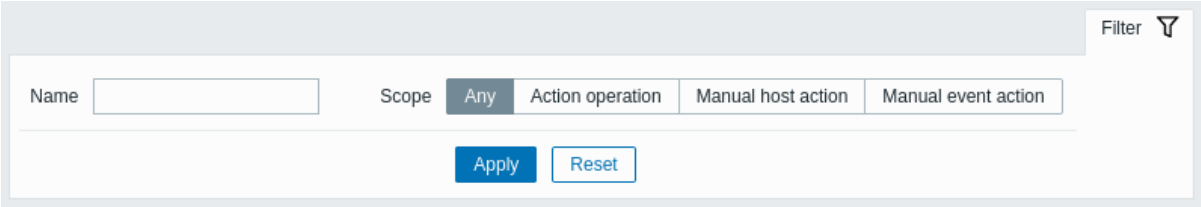
- *Usuń* - usuń skrypty

Aby użyć tej opcji, zaznacz pola wyboru przed odpowiednimi skryptami i kliknij *Usuń*.

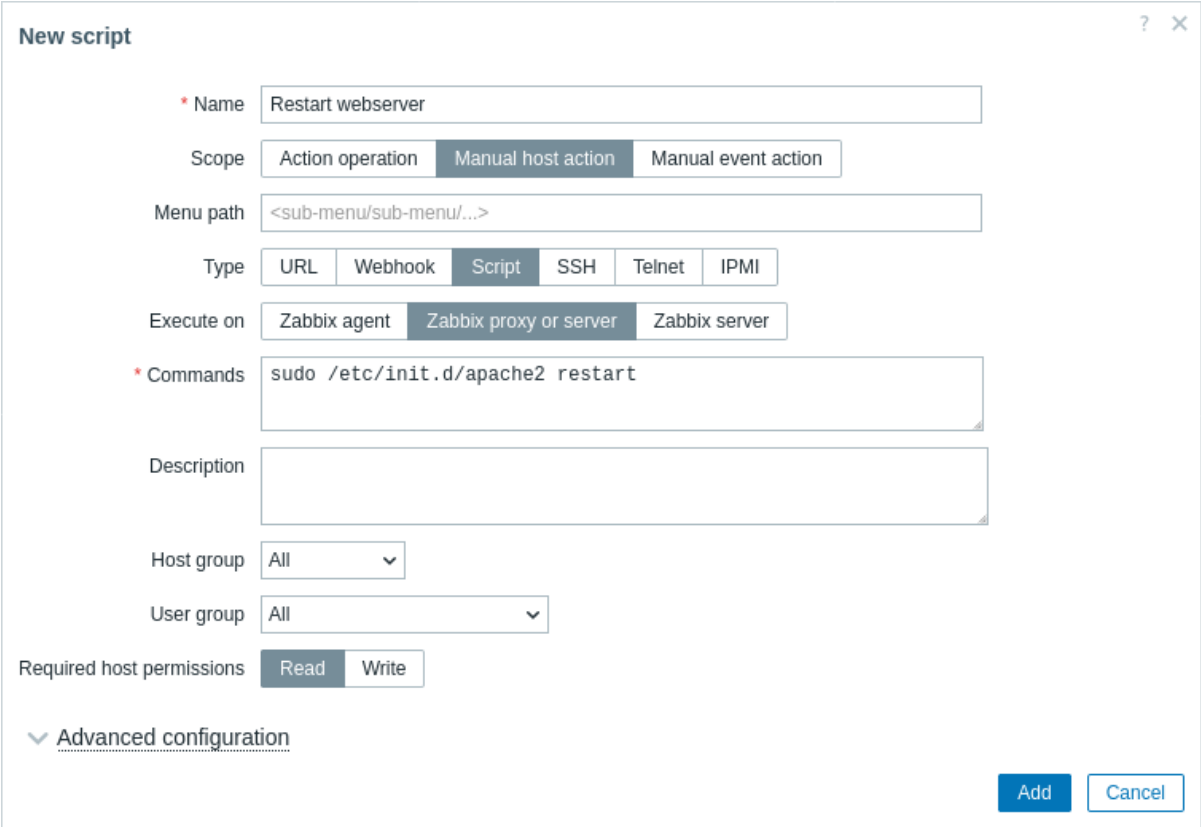
Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko interesujące Cię skrypty. Aby poprawić wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Łącze *Filter* jest dostępne nad listą skryptów. Po jego kliknięciu pojawi się filtr, za pomocą którego możesz filtrować skrypty według nazwy i zakresu.



Konfigurowanie skryptu globalnego



Note:
Zaleca się używanie funkcji makr zamiast zwykłych makr w skryptach globalnych w celu zwiększenia bezpieczeństwa, ponieważ makra nie są automatycznie zabezpieczane.

Atrybuty skryptu:

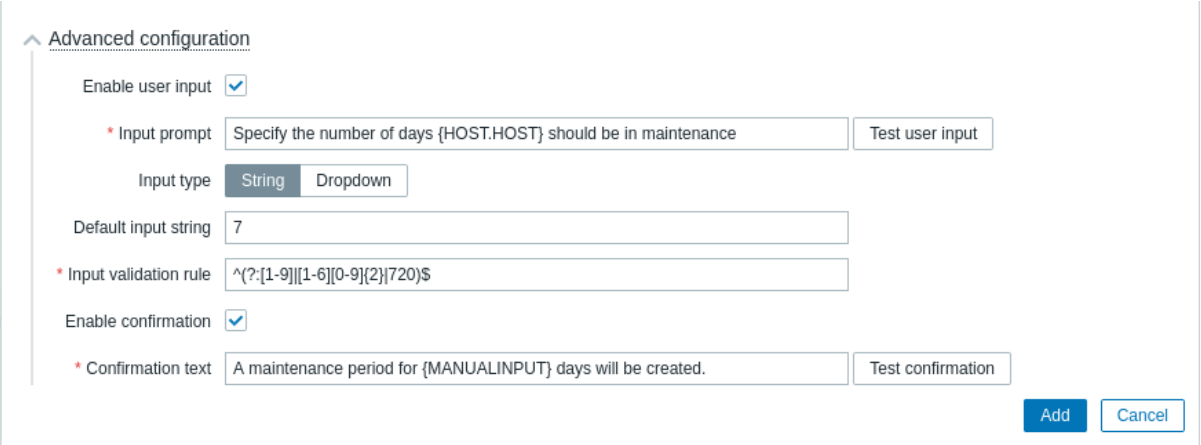
Parametr	Opis
Nazwa	Unikalna nazwa skryptu. Na przykład `Clear /tmp filesystem`

Parametr	Opis
Zakres	Zakres skryptu - operacja działania, ręczne działanie na hoście lub ręczne działanie na zdarzeniu. To ustawienie określa, gdzie można używać skryptu - w zdalnych poleceniach operacji działania, z menu hosta lub z menu zdarzenia. Ustawienie zakresu na „Action operation" udostępnia skrypt wszystkim użytkownikom mającym dostęp do *Alerts* > *Actions*. Jeśli skrypt jest faktycznie używany w działaniu, nie można zmienić jego zakresu z „action operation". **Obsługa makr** Zakres wpływa na zakres dostępnych makr. Na przykład makra związane z użytkownikiem (`{USER.*}`) są obsługiwane w skryptach, aby umożliwić przekazywanie informacji o użytkowniku, który uruchomił skrypt. Nie są jednak obsługiwane, jeśli zakres skryptu to operacja działania, ponieważ operacje działania są wykonywane automatycznie. Makro `{MANUALINPUT}` umożliwia określenie danych wejściowych podczas wykonywania skryptu. Jest obsługiwane w skryptach ręcznego działania na hoście i ręcznego działania na zdarzeniu. Aby dowiedzieć się, które inne makra są obsługiwane, wyszukaj „Trigger-based notifications and commands/Trigger-based commands", „Manual host action scripts" oraz „Manual event action scripts" w tabeli obsługiwanych makr. Pamiętaj, że jeśli makro może zostać rozwinięte do wartości zawierającej spacje (na przykład nazwa hosta), należy je odpowiednio ująć w cudzysłowy.
Ścieżka menu	Żądana ścieżka menu skryptu. Na przykład `Default` lub `Default/` spowoduje wyświetlenie skryptu w odpowiednim katalogu. Menu mogą być zagnieżdżane, np. `Main menu/Sub menu1/Sub menu2`. Podczas uzyskiwania dostępu do skryptów za pośrednictwem menu hosta/zdarzenia w sekcjach monitorowania będą one uporządkowane zgodnie z podanymi katalogami. To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Manual host action" lub „Manual event action".
Typ	Kliknij odpowiedni przycisk, aby wybrać typ skryptu: **URL**, **Webhook**, **Script**, **SSH**, **Telnet** lub polecenie **IPMI**. Typ **URL** jest dostępny tylko wtedy, gdy jako *Zakres* wybrano „Manual host action" lub „Manual event action".
Typ skryptu: **URL**	
URL	Określ adres URL umożliwiający szybki dostęp z menu hosta lub menu zdarzenia. Obsługiwane są makra i niestandardowe makra użytkownika. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej). Użyj makra {MANUALINPUT} w tym polu, aby móc określić dane wejściowe podczas wykonywania skryptu, na przykład: `http://{MANUALINPUT}/zabbix/zabbix.php?action=dashboard.view` Wartości makr nie mogą być kodowane w adresie URL.
Otwórz w nowym oknie	Określa, czy adres URL ma zostać otwarty w nowej, czy w tej samej karcie przeglądarki.
Typ skryptu: **Webhook**	
Parametry	Określ zmienne webhooka jako pary atrybut-wartość. Zobacz także: konfiguracja mediów typu Webhook. W wartościach parametrów obsługiwane są makra i niestandardowe makra użytkownika. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej).
Skrypt	Wprowadź kod JavaScript w edytorze modalnym, który otwiera się po kliknięciu pola parametru lub ikony ołówka obok niego. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej). Zobacz także: konfiguracja mediów typu Webhook, Dodatkowe obiekty JavaScript.
Limit czasu	Limit czasu wykonywania JavaScript (1-60 s, domyślnie 30 s). Obsługiwane są przyrostki czasu (np. `30s`, `1m`).
Typ skryptu: **Script**	

Parametr	Opis
Wykonaj na	Kliknij odpowiedni przycisk, aby wykonać skrypt powłoki na: **Zabbix agent** - skrypt zostanie wykonany przez Zabbix agent (jeśli pozycja system.run jest dozwolona) na hoście **Zabbix proxy lub serwerze** - skrypt zostanie wykonany przez Zabbix proxy lub serwer, zależnie od tego, czy host jest monitorowany przez proxy czy serwer. Zostanie wykonany na proxy, jeśli włączono tę funkcję za pomocą EnableRemoteCommands. Zostanie wykonany na serwerze, jeśli skrypty globalne są włączone za pomocą parametru serwera EnableGlobalScripts. **Zabbix server** - skrypt zostanie wykonany wyłącznie przez Zabbix server. Ta opcja nie będzie dostępna, jeśli skrypty globalne są wyłączone za pomocą parametru serwera EnableGlobalScripts.
Polecenia	Wprowadź pełną ścieżkę do poleceń, które mają zostać wykonane w ramach skryptu. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Typ skryptu: **SSH**	
Metoda uwierzytelniania	Wybierz metodę uwierzytelniania - hasło lub klucz publiczny.
Nazwa użytkownika	Wprowadź nazwę użytkownika.
Hasło	Wprowadź hasło. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Password”.
Plik klucza publicznego	Wprowadź ścieżkę do pliku klucza publicznego. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Public key”.
Plik klucza prywatnego	Wprowadź ścieżkę do pliku klucza prywatnego. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Public key”.
Hasło klucza	Wprowadź hasło klucza. To pole jest dostępne, jeśli jako metodę uwierzytelniania wybrano „Public key”.
Port	Wprowadź port zdalnej usługi SSH na docelowym hoście, z którym Zabbix nawiąże połączenie.
Polecenia	Wprowadź polecenia. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Typ skryptu: **Telnet**	
Nazwa użytkownika	Wprowadź nazwę użytkownika.
Hasło	Wprowadź hasło.
Port	Wprowadź port zdalnej usługi Telnet na docelowym hoście, z którym Zabbix nawiąże połączenie.
Polecenia	Wprowadź polecenia. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Typ skryptu: **IPMI**	
Polecenie	Wprowadź polecenie IPMI. Obsługa makr zależy od zakresu skryptu (zobacz *Zakres* powyżej). Obsługiwane są niestandardowe makra użytkownika.
Opis	Wprowadź opis skryptu.
Grupa hostów	Wybierz grupę hostów, dla której skrypt będzie dostępny (lub *All* dla wszystkich grup hostów).
Grupa użytkowników	Wybierz grupę użytkowników, dla której skrypt będzie dostępny (lub *All* dla wszystkich grup użytkowników). To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Manual host action” lub „Manual event action”.
Wymagane uprawnienia do hosta	Wybierz poziom uprawnień dla grupy hostów - *Read* lub *Write*. Tylko użytkownicy z wymaganym poziomem uprawnień będą mieć dostęp do wykonywania skryptu. To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Manual host action” lub „Manual event action”.
Konfiguracja zaawansowana	Kliknij nagłówek *Konfiguracja zaawansowana*, aby wyświetlić opcje konfiguracji zaawansowanej. To pole jest wyświetlane tylko wtedy, gdy jako *Zakres* wybrano „Manual host action” lub „Manual event action”.

Konfiguracja zaawansowana

Opcje konfiguracji zaawansowanej są dostępne w zwijanej sekcji *Konfiguracja zaawansowana*:



Parametr	Opis
Włącz dane wejściowe użytkownika	Zaznacz pole wyboru, aby włączyć ręczne wprowadzanie danych przez użytkownika przed wykonaniem skryptu. Ręcznie wprowadzone dane użytkownika zastąpią wartość makra {MANUALINPUT} w skrypcie. Zobacz także: Ręczne dane wejściowe użytkownika.
Monit o dane wejściowe	Wprowadź niestandardowy tekst zachęcający do wprowadzenia danych przez użytkownika. Ten tekst zostanie wyświetlony nad polem wprowadzania w wyskakującym oknie *Ręczne dane wejściowe*. Aby wyświetlić podgląd wyskakującego okna *Ręczne dane wejściowe*, kliknij *Testuj dane wejściowe użytkownika*. Podgląd umożliwia także sprawdzenie, czy ciąg wejściowy jest zgodny z regułą walidacji danych wejściowych (zobacz poniższe parametry). Obsługa makr i makr użytkownika zależy od zakresu skryptu (zobacz *Zakres* w ogólnych parametrach konfiguracji skryptu).
Typ danych wejściowych	Wybierz typ ręcznie wprowadzanych danych: **Ciąg** - pojedynczy ciąg; **Lista rozwijana** - wartość jest wybierana spośród wielu opcji listy rozwijanej.
Opcje listy rozwijanej	Wprowadź unikatowe wartości listy rozwijanej danych wejściowych użytkownika w postaci listy rozdzielanej przecinkami. Aby uwzględnić pustą opcję na liście rozwijanej, dodaj dodatkowy przecinek na początku, w środku lub na końcu listy. To pole jest wyświetlane tylko wtedy, gdy jako *Typ danych wejściowych* wybrano opcję „Lista rozwijana”.
Domyślny ciąg wejściowy	Wprowadź domyślny ciąg danych wejściowych użytkownika (lub pozostaw puste). To pole zostanie zweryfikowane za pomocą wyrażenia regularnego podanego w polu *Reguła walidacji danych wejściowych*. Wprowadzona tutaj wartość będzie domyślnie wyświetlana w wyskakującym oknie *Ręczne dane wejściowe*. To pole jest wyświetlane tylko wtedy, gdy jako *Typ danych wejściowych* wybrano opcję „Ciąg”.
Reguła walidacji danych wejściowych	Wprowadź wyrażenie regularne w celu zweryfikowania ciągu danych wejściowych użytkownika. Obsługiwane są globalne wyrażenia regularne. To pole jest wyświetlane tylko wtedy, gdy jako *Typ danych wejściowych* wybrano opcję „Ciąg”.
Włącz potwierdzenie	Zaznacz pole wyboru, aby wyświetlać komunikat potwierdzenia przed wykonaniem skryptu. Ta funkcja może być szczególnie przydatna w przypadku potencjalnie niebezpiecznych operacji (takich jak skrypt ponownego uruchomienia) lub operacji, których wykonanie może zająć dużo czasu.
Tekst potwierdzenia	Wprowadź niestandardowy tekst potwierdzenia dla wyskakującego okna potwierdzenia włączonego za pomocą powyższego pola wyboru (na przykład *System zdalny zostanie uruchomiony ponownie. Czy na pewno?*). Aby zobaczyć, jak będzie wyglądał tekst, kliknij *Testuj potwierdzenie* obok pola. Obsługiwane są makra i niestandardowe makra użytkownika. *Uwaga:* makra nie zostaną rozwinięte podczas testowania komunikatu potwierdzenia.

Jeśli skonfigurowano zarówno ręczne dane wejściowe użytkownika, jak i komunikat potwierdzenia, zostaną one wyświetlone w kolejnych wyskakujących oknach.

Ręczne wprowadzanie danych przez użytkownika

Ręczne wprowadzanie danych przez użytkownika umożliwia podanie niestandardowego parametru przy każdym wykonaniu skryptu. Eliminuje to konieczność tworzenia wielu podobnych skryptów użytkownika, różniących się tylko jednym parametrem.

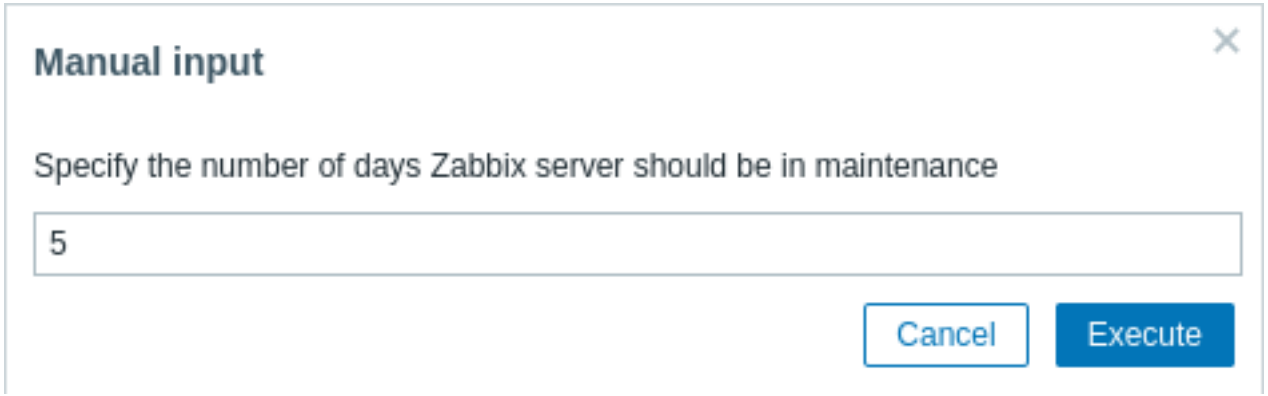
Na przykład podczas wykonywania skryptu można chcieć podać inną liczbę całkowitą lub inny adres URL.

Aby włączyć ręczne wprowadzanie danych przez użytkownika:

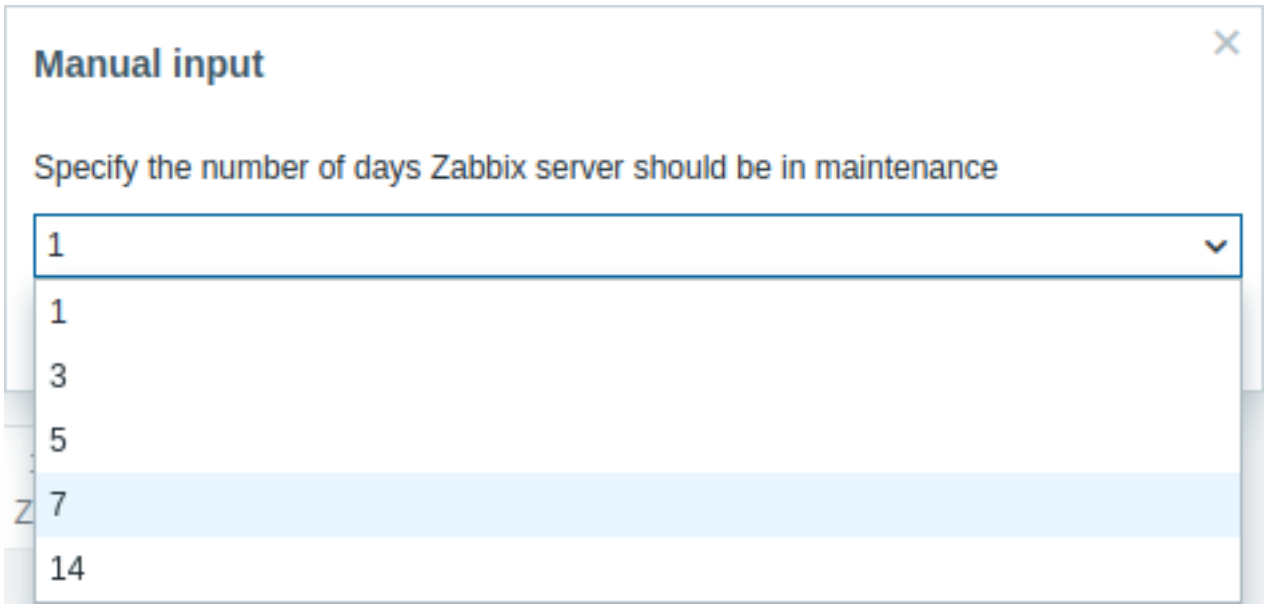
- użyj makra {MANUALINPUT} w skrypcie (polecenia, skrypt, parametr skryptu) w wymaganym miejscu lub w polu URL skryptów URL;
- w zaawansowanej konfiguracji skryptu włącz ręczne wprowadzanie danych przez użytkownika i skonfiguruj opcje wprowadzania.

Po włączeniu wprowadzania danych przez użytkownika przed wykonaniem skryptu zostanie wyświetlone użytkownikowi okno podręczne *Ręczne wprowadzanie danych* z prośbą o podanie niestandardowej wartości. Podana wartość zastąpi {MANUALINPUT} w skrypcie.

W zależności od konfiguracji użytkownik zostanie poproszony o wprowadzenie wartości tekstowej:



Lub wybranie wartości z listy rozwijanej zawierającej wstępnie zdefiniowane opcje:



Ręczne wprowadzanie danych przez użytkownika jest dostępne tylko dla skryptów, których zakres to „Ręczne działanie na hoście" lub „Ręczne działanie na zdarzeniu".

Wykonywanie skryptu i wynik

Skrypty uruchamiane przez serwer Zabbix są wykonywane w kolejności opisanej na stronie wykonywania poleceń.

Wynik skryptu jest wyświetlany w wyskakującym oknie, które pojawia się po uruchomieniu skryptu. Wartością zwracaną skryptu jest standardowe wyjście:

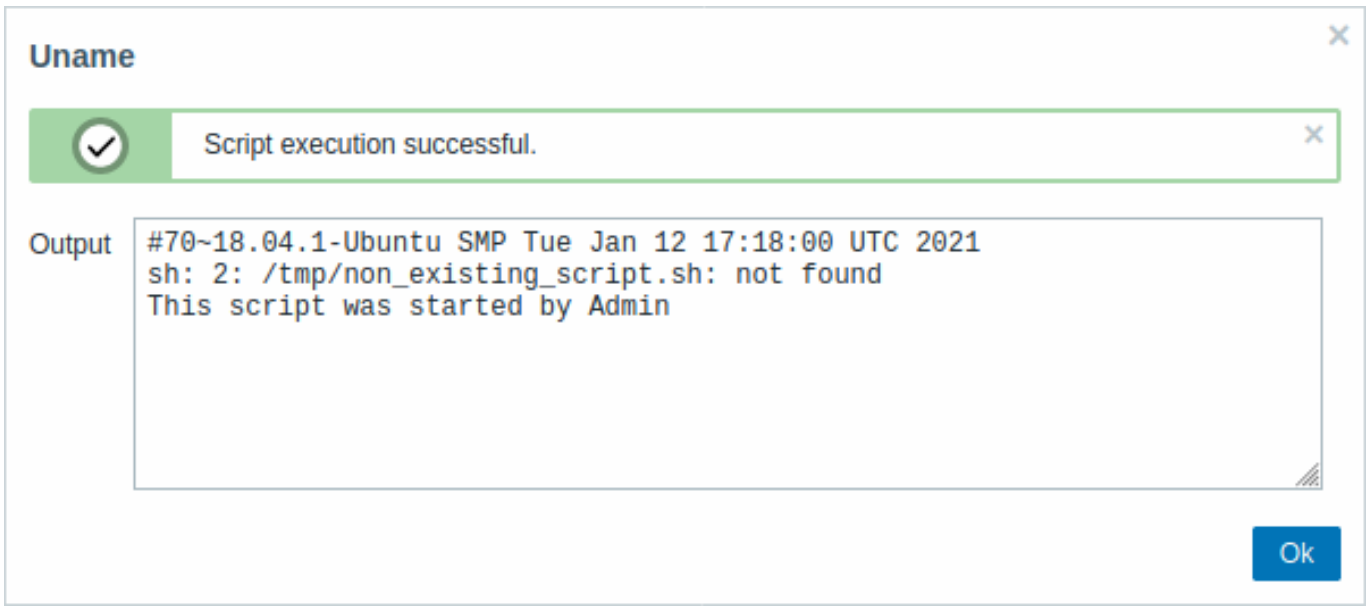
- Jeśli skrypt zakończy się pomyślnie (kod wyjścia 0), wartość zwracana jest ograniczona do 16 MB (włącznie z końcowymi białymi znakami, które są obcinane).
- Jeśli skrypt zakończy się błędem (niezerowy kod wyjścia), wartością zwracaną jest standardowe wyjście błędów ograniczone do 2 KB.

Domyślnie Zabbix nie przechowuje rozszerzonego wyjścia skryptu. Aby zachować pełne szczegóły wyjścia, można zaimplementować rejestrowanie w samym skrypcie (np. przekierować wyjście do lokalnego pliku dziennika).

Należy pamiętać, że w przypadku skryptów wykonywanych zarówno na serwerze Zabbix, jak i na Zabbix proxy, obowiązują również ograniczenia bazy danych.

Poniżej znajduje się przykład skryptu i okna wyniku:

```
uname -v
/tmp/non_existing_script.sh
echo "This script was started by {USER.USERNAME}"
```



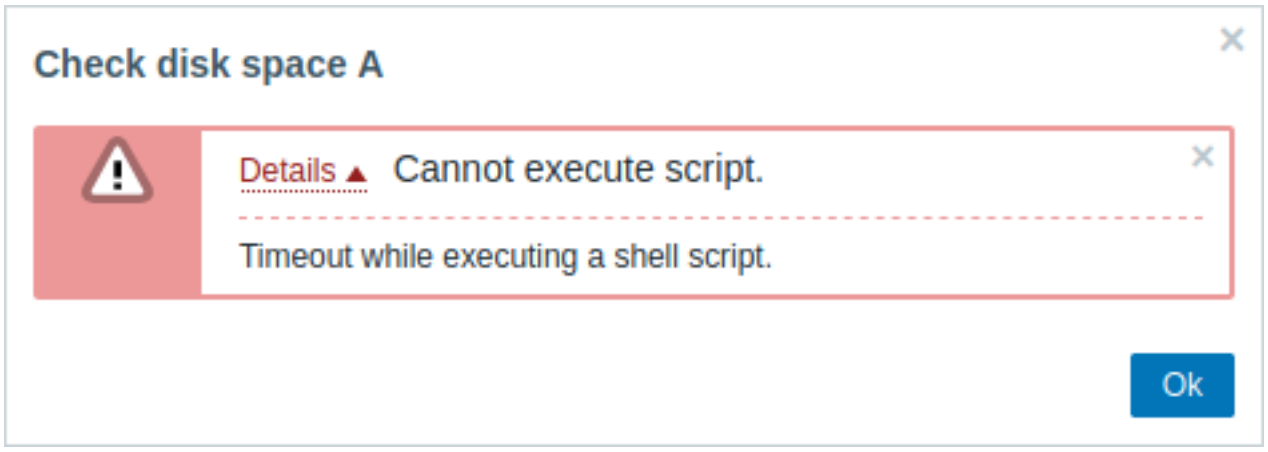
Wynik skryptu nie wyświetla samego skryptu.

Przekroczenie limitu czasu skryptu na agencie Zabbix

Może wystąpić sytuacja, w której podczas wykonywania skryptu zostanie przekroczony limit czasu.

Poniżej przedstawiono przykład skryptu uruchamianego na agencie Zabbix oraz okno z wynikiem:

```
sleep 5
df -h
```



W tym przypadku wyświetlony zostanie następujący komunikat o błędzie:

```
Timeout while executing a shell script.
```

Aby uniknąć takich sytuacji, zaleca się zoptymalizowanie samego skryptu (w powyższym przykładzie wartości „5”) zamiast dostosowywania parametru `Timeout` w konfiguracji agenta Zabbix i konfiguracji serwera Zabbix.

Jeśli parametr `Timeout` został zmieniony w konfiguracji agenta Zabbix, zostanie wyświetlony następujący komunikat o błędzie:

```
Get value from agent failed: ZBX_TCP_READ() timed out.
```

Oznacza to, że modyfikacja została wprowadzona w konfiguracji agenta Zabbix, ale konieczne jest również zmodyfikowanie parametru `Timeout` w konfiguracji serwera Zabbix.

Limit czasu skryptu na agentach aktywnych

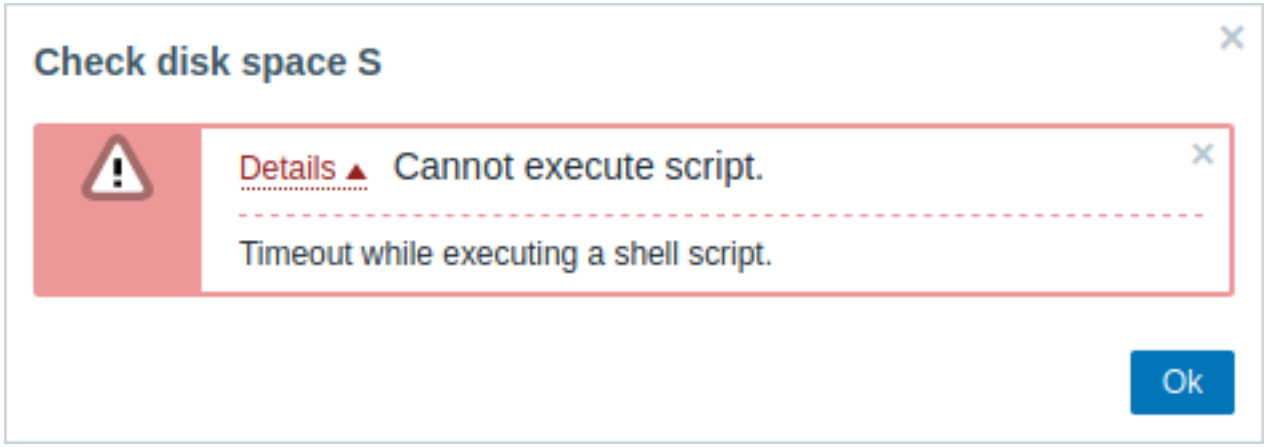
Możliwe jest wykonywanie skryptów na agentach działających wyłącznie w trybie aktywnym. Gdy wykonanie skryptu zostanie wyzwolone przez operację akcji lub ręczne uruchomienie skryptu, polecenie jest dołączane do konfiguracji aktywnego sprawdzania i wykonywane, gdy aktywny agent je otrzyma.

Skrypty uruchamiane ręcznie są wysyłane do aktywnego agenta wraz z limitem czasu serwer/proxy dla wykonania skryptu. Zwiększ domyślny limit czasu serwer/proxy dla wykonania skryptu. Limit czasu musi być większy niż częstotliwość odświeżania aktywnego sprawdzania, w przeciwnym razie limit czasu zostanie przekroczony, zanim aktywny agent otrzyma skrypt i będzie mógł zwrócić wynik.

Limit czasu skryptu na serwerze/proxy Zabbix

Poniżej przedstawiono przykład skryptu uruchamianego na serwerze Zabbix oraz okno wyników:

```
sleep 11
df -h
```



Zaleca się również zoptymalizowanie samego skryptu (zamiast dostosowywania parametru `TrapperTimeout` do odpowiedniej wartości (w naszym przypadku > 11) poprzez modyfikację konfiguracji serwera Zabbix).

Użytkownicy

Przegląd

To menu zawiera sekcje związane z konfiguracją użytkowników w Zabbix. To menu jest dostępne tylko dla użytkowników typu SuperAdmin.

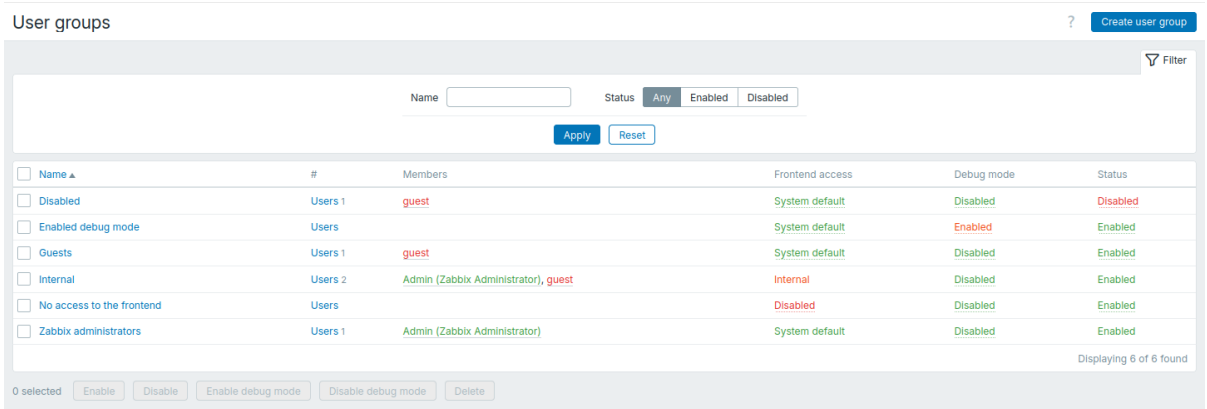
Grupy użytkowników

Przegląd

W sekcji *Users* > *User groups* zarządzane są grupy użytkowników systemu.

Grupy użytkowników

Wyświetlana jest lista istniejących grup użytkowników wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy użytkowników. Kliknięcie nazwy grupy użytkowników otwiera formularz konfiguracji grupy użytkowników.
#	Liczba użytkowników w grupie. Kliknięcie pozycji *Użytkownicy* wyświetli odpowiednich użytkowników odfiltrowanych na liście użytkowników.
Członkowie	Nazwy użytkowników należących do grupy użytkowników (w nawiasach podane są imię i nazwisko). Kliknięcie nazwy użytkownika otwiera formularz konfiguracji użytkownika. Użytkownicy z wyłączonych grup są wyświetlani na czerwono.
Dostęp do frontendu	Wyświetlany jest poziom dostępu do frontendu: **Domyślne ustawienie systemowe** - użytkownicy są uwierzytelniani przez Zabbix, LDAP lub HTTP (w zależności od globalnie ustawionej metody uwierzytelniania); **Wewnętrzne** - użytkownicy są uwierzytelniani przez Zabbix; ignorowane, jeśli globalnie ustawiono uwierzytelnianie HTTP; **LDAP** - użytkownicy są uwierzytelniani przez LDAP; ignorowane, jeśli globalnie ustawiono uwierzytelnianie HTTP; **Wyłączone** - dostęp do frontendu Zabbix jest zabroniony dla tej grupy. Klikając bieżący poziom, można go zmienić.
Tryb debugowania	Wyświetlany jest stan trybu debugowania - *Włączony* lub *Wyłączony*. Klikając stan, można go zmienić.
Stan	Wyświetlany jest stan grupy użytkowników - *Włączona* lub *Wyłączona*. Klikając stan, można go zmienić.

Aby skonfigurować nową grupę użytkowników, kliknij przycisk *Utwórz grupę użytkowników* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

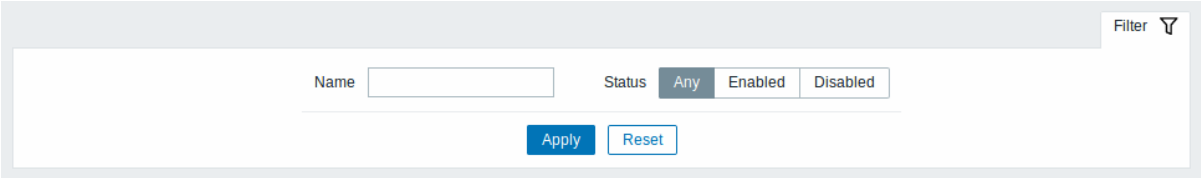
- *Włącz* - zmienia status grupy użytkowników na *Włączony*
- *Wyłącz* - zmienia status grupy użytkowników na *Wyłączony*
- *Włącz tryb debugowania* - włącza tryb debugowania dla grup użytkowników
- *Wyłącz tryb debugowania* - wyłącza tryb debugowania dla grup użytkowników
- *Usuń* - usuwa grupy użytkowników

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi grupami użytkowników, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te grupy użytkowników, które Cię interesują. Aby zapewnić lepszą wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

Łącze *Filter* jest dostępne nad listą grup użytkowników. Po kliknięciu go pojawi się filtr, za pomocą którego można filtrować grupy użytkowników według nazwy i statusu.



Role użytkownika

Przegląd

W sekcji *Użytkownicy* > *Role użytkowników* można tworzyć role użytkowników.

Role użytkowników umożliwiają tworzenie szczegółowych uprawnień na podstawie początkowo wybranego typu użytkownika (*Użytkownik*, *Administrator*, *Superadministrator*).

Po wybraniu typu użytkownika przyznawane są wszystkie dostępne dla niego uprawnienia (domyślnie zaznaczone).

Uprawnienia można odbierać tylko z podzbioru uprawnień dostępnych dla danego typu użytkownika; nie można ich rozszerzać poza zakres dostępny dla tego typu użytkownika.

Pola wyboru niedostępnych uprawnień są wyszarzone; użytkownicy nie będą mogli uzyskać dostępu do danego elementu nawet po wpisaniu bezpośredniego adresu URL tego elementu w przeglądarce.

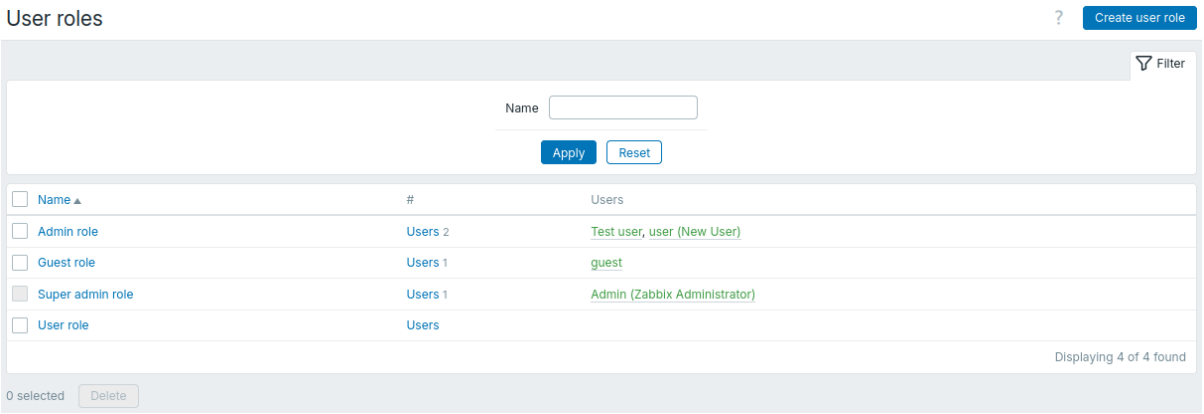
Note:
Ograniczenie dostępu do niektórych elementów interfejsu użytkownika zapobiega jedynie otwarciu danej strony — nie odbiera możliwości dostępu do danych bazowych w innych częściach interfejsu.

Role użytkowników można przypisywać użytkownikom systemu. Każdy użytkownik może mieć przypisaną tylko jedną rolę.

Domyślne role użytkowników

Domyślnie Zabbix jest skonfigurowany z czterema rolami użytkowników, które mają wstępnie zdefiniowany zestaw uprawnień:

- Rola Gość
- Rola Użytkownik
- Rola Administrator
- Rola Super administrator



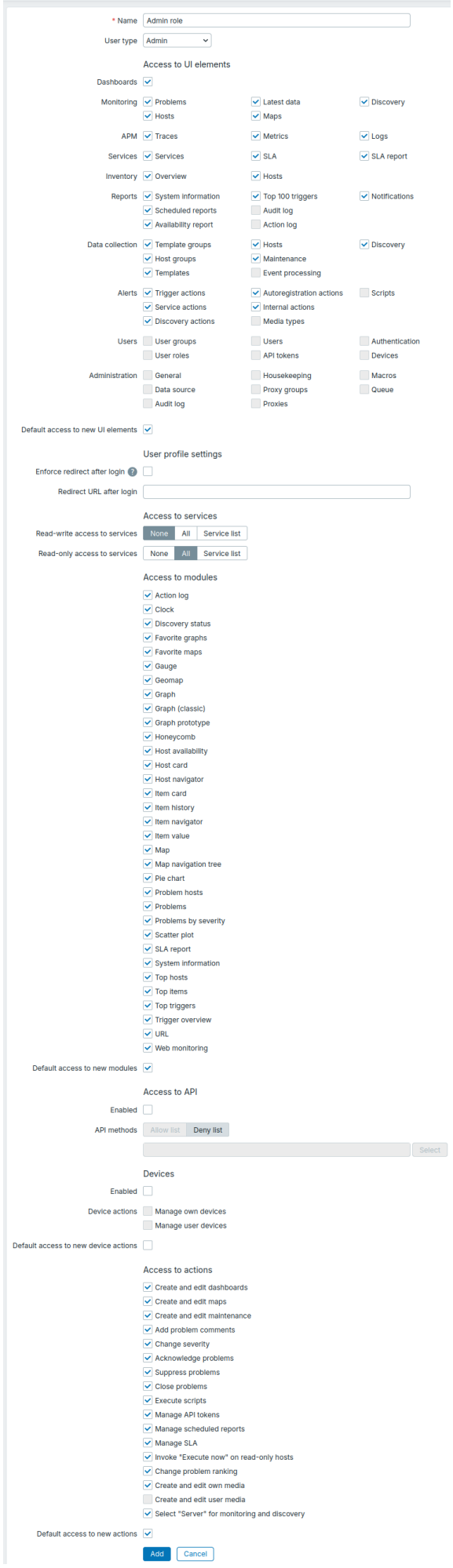
Role te są oparte na głównych typach użytkowników w Zabbix. Wyświetlana jest lista wszystkich użytkowników, którym przypisano daną rolę. Użytkownicy należący do wyłączonych grup są oznaczeni na czerwono. *Rola Gość* to rola typu użytkownika, która zapewnia wyłącznie uprawnienia do wyświetlania niektórych sekcji frontendu.

Note:
Domyślnej *roli Super administrator* nie można modyfikować ani usuwać, ponieważ w Zabbix musi istnieć co najmniej jeden użytkownik typu Super administrator z nieograniczonymi uprawnieniami. Użytkownicy typu *Super administrator* mogą modyfikować ustawienia własnej roli, ale nie mogą zmieniać typu użytkownika.

Konfiguracja

Aby utworzyć nową rolę, kliknij przycisk *Utwórz rolę użytkownika* w prawym górnym rogu. Aby zaktualizować istniejącą rolę, kliknij jej nazwę, aby otworzyć formularz konfiguracji.

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

Wyświetlone zostaną dostępne uprawnienia. Aby odebrać określone uprawnienie, odznacz jego pole wyboru.

Poniżej opisano dostępne uprawnienia wraz z wartościami domyślnymi dla każdej wstępnie skonfigurowanej roli użytkownika w Zabbix.

Domyślne uprawnienia

Dostęp do elementów interfejsu

Domyślny dostęp do sekcji menu zależy od typu użytkownika. Szczegółowe informacje można znaleźć na stronie Uprawnienia details.

Dostęp do innych opcji

Niektóre sekcje opisane w poniższej tabeli mogą być ukryte:

- *Dostęp do modułów* jest ukryty, jeśli parametr `$ZBX_FEATURE_FLAGS['modules_config_enabled']` ma wartość `false` w pliku konfiguracji frontendu.
- *Urządzenia* są ukryte, jeśli parametr `EnableMobileDevices` jest wyłączony.

Parametr	**Opis**	**Domyślne role użytkowników**			
		Rola superadministratora	*Rola administratora*	*Rola użytkownika*	*Rola gościa*
Domyślny dostęp do nowych elementów interfejsu	Ta opcja określa, jak nowe sekcje menu będą dostępne po aktualizacji Zabbix. Istniejące sekcje menu modułów pozostają bez zmian.	Tak	Tak	Tak	Tak
Ustawienia profilu użytkownika					
Wymuś przekierowanie po zalogowaniu	Zaznacz pole wyboru, aby ograniczyć użytkowników do adresu URL przekierowania po zalogowaniu. Jeśli pole nie jest zaznaczone, użytkownicy mogą ustawić własny adres URL po zalogowaniu w swoim profilu; adres URL przekierowania po zalogowaniu jest używany tylko jako rozwiązanie awaryjne.	Tak	Nie	Nie	Nie
Adres URL przekierowania po zalogowaniu	Określ adres URL przekierowania po zalogowaniu.				
Dostęp do usług					

Dostęp do usług z uprawnieniami odczytu i zapisu	Wybierz dostęp do usług z uprawnieniami odczytu i zapisu: **Brak** - całkowity brak dostępu **Wszystkie** - dostęp do wszystkich usług z uprawnieniami odczytu i zapisu **Lista usług** - wybierz usługi z dostępem do odczytu i zapisu Przyznany dostęp do odczytu i zapisu ma pierwszeństwo przed ustawieniami dostępu tylko do odczytu i jest dynamicznie dziedziczony przez usługi podrzędne.	Wszystkie	Wszystkie	Brak	Brak
Dostęp do usług z uprawnieniami odczytu i zapisu według znacznika	Określ nazwę znacznika i opcjonalnie jego wartość, aby dodatkowo przyznać dostęp do odczytu i zapisu usługom pasującym do znacznika. Ta opcja jest dostępna, jeśli w parametrze *Dostęp do usług z uprawnieniami odczytu i zapisu* wybrano opcję *Lista usług*. Przyznany dostęp do odczytu i zapisu ma pierwszeństwo przed ustawieniami dostępu tylko do odczytu i jest dynamicznie dziedziczony przez usługi podrzędne.				
Dostęp do usług tylko do odczytu	Wybierz dostęp tylko do odczytu do usług: **Brak** - całkowity brak dostępu **Wszystkie** - dostęp tylko do odczytu do wszystkich usług **Lista usług** - wybierz usługi z dostępem tylko do odczytu Dostęp tylko do odczytu nie ma pierwszeństwa przed dostępem do odczytu i zapisu i jest dynamicznie dziedziczony przez usługi podrzędne.			Wszystkie	Wszystkie
Dostęp tylko do odczytu do usług według znacznika	Określ nazwę znacznika i opcjonalnie jego wartość, aby dodatkowo przyznać dostęp tylko do odczytu usługom pasującym do znacznika. Ta opcja jest dostępna, jeśli w parametrze *Dostęp do usług tylko do odczytu* wybrano opcję *Lista usług*. Dostęp tylko do odczytu nie ma pierwszeństwa przed dostępem do odczytu i zapisu i jest dynamicznie dziedziczony przez usługi podrzędne.				
Dostęp do modułów					
<Nazwa modułu>	Zezwól na dostęp do określonego modułu lub go zabroń. W tej sekcji wyświetlane są tylko włączone moduły. Nie można przyznać ani ograniczyć dostępu do modułu, który jest obecnie wyłączony.	Tak	Tak	Tak	Tak
Domyślny dostęp do nowych modułów	Ta opcja określa, jak nowe moduły i widżety będą dostępne po aktualizacji Zabbix. Dotyczy również modułów i widżetów dodanych w sekcji *Administracja > Ogólne > Moduły*.				
Dostęp do API					
Włączone	Włącz lub wyłącz dostęp do API.	Tak	Nie	Nie	Nie

Metody API	Wybierz opcję *Lista dozwolonych*, aby zezwolić na metody API określone w polu wyszukiwania, lub *Lista zabronionych*, aby ich zabronić. Należy pamiętać, że nie można jednocześnie zezwolić na niektóre metody API i zabronić innych. W polu wyszukiwania zacznij wpisywać nazwę metody, a następnie wybierz ją z listy autouzupełniania. Możesz również kliknąć przycisk *Wybierz* i wybrać metody z pełnej listy dostępnej dla tego typu użytkownika. Należy pamiętać, że jeśli określone działanie z bloku Dostęp do działań nie jest zaznaczone, użytkownicy nie będą mogli używać metod API związanych z tym działaniem. Obsługiwane są symbole wieloznaczne. Przykłady: `dashboard.*` (wszystkie metody usługi API „dashboard.”), `*` (dowolna metoda), `*.export` (metody o nazwie „.export” ze wszystkich usług API). Jeśli nie określono żadnych metod, *Lista dozwolonych* zabrania dostępu do wszystkich metod API, natomiast *Lista zabronionych* zezwala na dostęp do wszystkich metod API dostępnych dla danego typu użytkownika. Aby zezwolić na wszystkie metody za pomocą opcji *Lista dozwolonych*, określ `*`. Aby całkowicie zablokować dostęp do API, wyczyść pole wyboru *Włączone* zamiast pozostawiać pustą *Listę dozwolonych*. Lista metod API nie dotyczy metod, które nie wymagają uwierzytelniania, takich jak `apiinfo.version` i `user.login`.				
Dostęp do działań					
Tworzenie i edytowanie pulpitów	Wyczyszczenie tego pola wyboru spowoduje również odebranie uprawnień do używania metod API `.create`, `.update` i `.delete` dla odpowiednich elementów.	Tak	Tak	Tak	Nie
Tworzenie i edytowanie map					
Tworzenie i edytowanie okresów utrzymania				Nie	
Dodawanie komentarzy do problemów	Wyczyszczenie tego pola wyboru spowoduje również odebranie uprawnień do wykonywania odpowiedniego działania za pomocą metody API `event.acknowledge`.			Tak	
Zmiana ważności					
Potwierdzanie problemów					
Wyciszanie problemów					
Zamykanie problemów					
Wykonywanie skryptów	Wyczyszczenie tego pola wyboru spowoduje również odebranie uprawnień do używania metody API `script.execute`.				

Zarządzanie tokenami API	Wyczyszczenie tego pola wyboru spowoduje również odebranie uprawnień do używania wszystkich metod API `token.*`.				
Zarządzanie raportami zaplanowanymi	Wyczyszczenie tego pola wyboru spowoduje również odebranie uprawnień do używania wszystkich metod API `report.*`.			Nie	
Zarządzanie SLA	Włącz lub wyłącz uprawnienia do zarządzania SLA.				
Wywoływanie opcji „Wykonaj teraz” na hostach tylko do odczytu	Zezwól użytkownikom na używanie opcji „Wykonaj teraz” w sekcji Najnowsze dane dla pozycji na hostach tylko do odczytu.			Tak	
Zmiana rankingu problemów	Zezwól użytkownikom na zmianę rankingu problemu z przyczyny na objaw i odwrotnie.				
Tworzenie i edytowanie własnych mediów	Zezwól użytkownikom na tworzenie i edytowanie własnych mediów.				
Tworzenie i edytowanie mediów użytkowników	Zezwól użytkownikom na tworzenie i edytowanie mediów innych użytkowników. Ta opcja jest dostępna tylko dla użytkowników będących superadministratorami.		Nie	Nie	
Wybór „Serwer” do monitorowania i wykrywania	Wyczyszczenie tego pola wyboru ogranicza tworzenie hostów monitorowanych przez serwer oraz reguł wykrywania sieci wykonywanych przez serwer, a także testowanie pozycji za pomocą serwera. Opcja *Serwer* będzie wyłączona w konfiguracji hosta dla parametru *Monitorowane przez*, w konfiguracji reguły wykrywania sieci dla parametru *Wykrywane przez* oraz w testowaniu pozycji dla parametru *Testuj za pomocą*.		Tak	Nie	
Domyślny dostęp do nowych działań	Ta opcja określa, jak nowe działania będą dostępne po aktualizacji Zabbix.		Tak	Tak	
Urządzenia					
Włączone	Włącz lub wyłącz dostęp do *Ustawienia użytkownika* > *Urządzenia* (dostępne dla wszystkich użytkowników) oraz *Użytkownicy* > *Urządzenia* (dostępne tylko dla superadministratorów).	Tak	Nie	Nie	Nie
Zarządzanie własnymi urządzeniami	Zezwól użytkownikom na zarządzanie własnymi urządzeniami w sekcji *Ustawienia użytkownika* > *Urządzenia*.				
Zarządzanie urządzeniami użytkowników	Zezwól użytkownikom na zarządzanie urządzeniami należącymi do innych użytkowników w sekcji *Użytkownicy* > *Urządzenia*.				
Domyślny dostęp do nowych działań	Ta opcja określa, jak nowe działania będą dostępne po aktualizacji Zabbix.				

Zobacz także:

- Konfigurowanie użytkownika

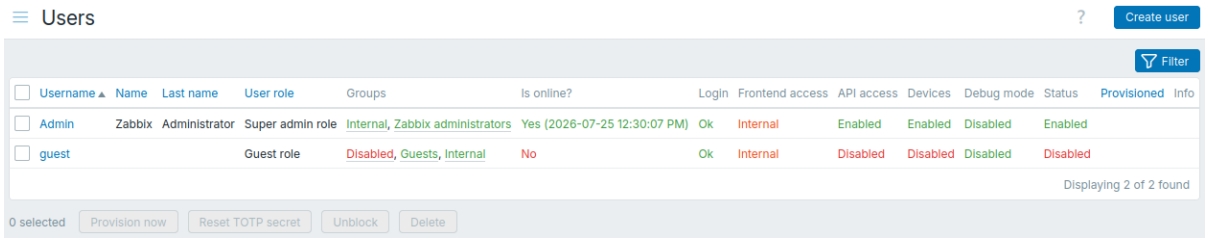
Użytkownicy

Przegląd

W sekcji *Użytkownicy* > *Użytkownicy* zarządza się użytkownikami systemu.

Użytkownicy

Wyświetlana jest lista istniejących użytkowników wraz z ich szczegółami.



Wyświetlane dane:

Kolumna	Opis
Nazwa użytkownika	Nazwa użytkownika używana do logowania się do Zabbix. Kliknięcie nazwy użytkownika otwiera formularz konfiguracji użytkownika.
Imię	Imię użytkownika.
Nazwisko	Nazwisko użytkownika.
Rola użytkownika	Wyświetlana jest rola użytkownika.
Grupy	Wyświetlane są grupy, których członkiem jest użytkownik. Kliknięcie nazwy grupy użytkowników otwiera formularz konfiguracji grupy użytkowników. Wyłączone grupy są wyświetlane na czerwono.
Czy jest online?	Wyświetlany jest status użytkownika online - *Tak* lub *Nie*. W nawiasach wyświetlany jest czas ostatniej aktywności użytkownika.
Logowanie	Wyświetlany jest status logowania użytkownika - *OK* lub *Zablokowane*. Użytkownik może zostać tymczasowo zablokowany po przekroczeniu liczby nieudanych prób logowania określonej w sekcji *Administracja > Ogólne > Inne* (domyślnie pięciu). Kliknięcie opcji *Zablokowane* odblokowuje użytkownika.
Dostęp do frontendu	Wyświetlany jest poziom dostępu do frontendu - *Domyślne ustawienia systemowe*, *Wewnętrzne*, *LDAP* lub *Wyłączone*, zależnie od ustawienia dla całej grupy użytkowników.
Dostęp do API	Wyświetlany jest status dostępu do API - *Włączony* lub *Wyłączony*, zależnie od ustawienia dla roli użytkownika. Jeśli dostęp do API jest włączony, umieszczenie wskaźnika myszy nad opcją *Włączony* wyświetla metody API roli użytkownika w sekcji *Dozwolone metody* lub *Zabronione metody*, zależnie od trybu metod API. Jeśli nie określono żadnych metod, wyświetlana jest wartość *Brak*.
Urządzenia	Wyświetlany jest status dostępu do urządzeń - *Włączony* lub *Wyłączony*, zależnie od ustawienia dla roli użytkownika. Ta kolumna jest wyświetlana, jeśli włączono `EnableMobileDevices`.
Tryb debugowania	Wyświetlany jest status trybu debugowania - *Włączony* lub *Wyłączony*, zależnie od ustawienia dla całej grupy użytkowników.
Status	Wyświetlany jest status użytkownika - *Włączony* lub *Wyłączony*, zależnie od ustawienia dla całej grupy użytkowników.
Utworzono automatycznie	Wyświetlana jest data ostatniego automatycznego utworzenia użytkownika. Dotyczy użytkowników utworzonych za pomocą aprowizacji JIT z LDAP/SAML.
Informacje	Wyświetlane są informacje o błędach. Żółte ostrzeżenie jest wyświetlane dla użytkowników bez grup użytkowników. Czerwone ostrzeżenie jest wyświetlane dla użytkowników bez ról oraz dla użytkowników bez ról i grup użytkowników.

Aby skonfigurować nowego użytkownika, kliknij przycisk *Utwórz użytkownika* w prawym górnym rogu.

Opcje edycji zbiorczej

Przyciski poniżej listy oferują następujące opcje edycji zbiorczej:

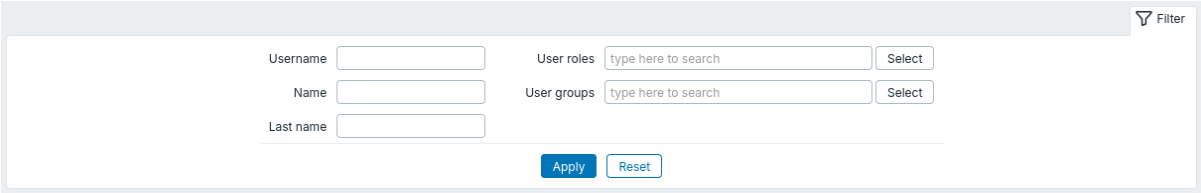
- *Provision now* - aktualizuje informacje o użytkowniku z LDAP (ta opcja jest dostępna tylko wtedy, gdy wybrano użytkownika LDAP)
- *Reset TOTP secret* - resetuje sekrety TOTP użytkowników dla wszystkich metod TOTP i usuwa sesję użytkownika (ta opcja jest dostępna tylko wtedy, gdy włączono MFA; w przypadku użytkowników bez sekretów TOTP ich sesja nie zostanie usunięta)
- *Unblock* - ponownie włącza dostęp do systemu zablokowanym użytkownikom
- *Delete* - usuwa użytkowników

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi użytkownikami, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko interesujących Cię użytkowników. W celu zwiększenia wydajności wyszukiwania dane są przeszukiwane z nierozwiniętymi makrami.

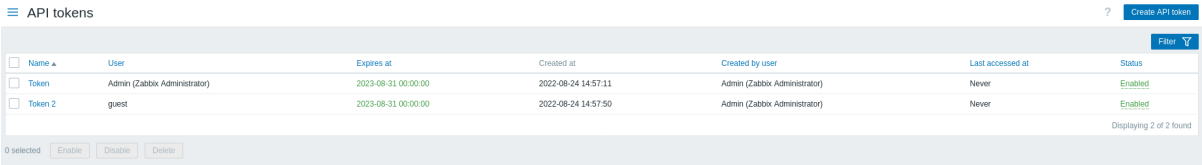
Link *Filtr* jest dostępny nad listą użytkowników. Po jego kliknięciu pojawi się filtr, za pomocą którego można filtrować użytkowników według nazwy użytkownika, imienia, nazwiska, roli użytkownika i grupy użytkowników.



Tokeny API

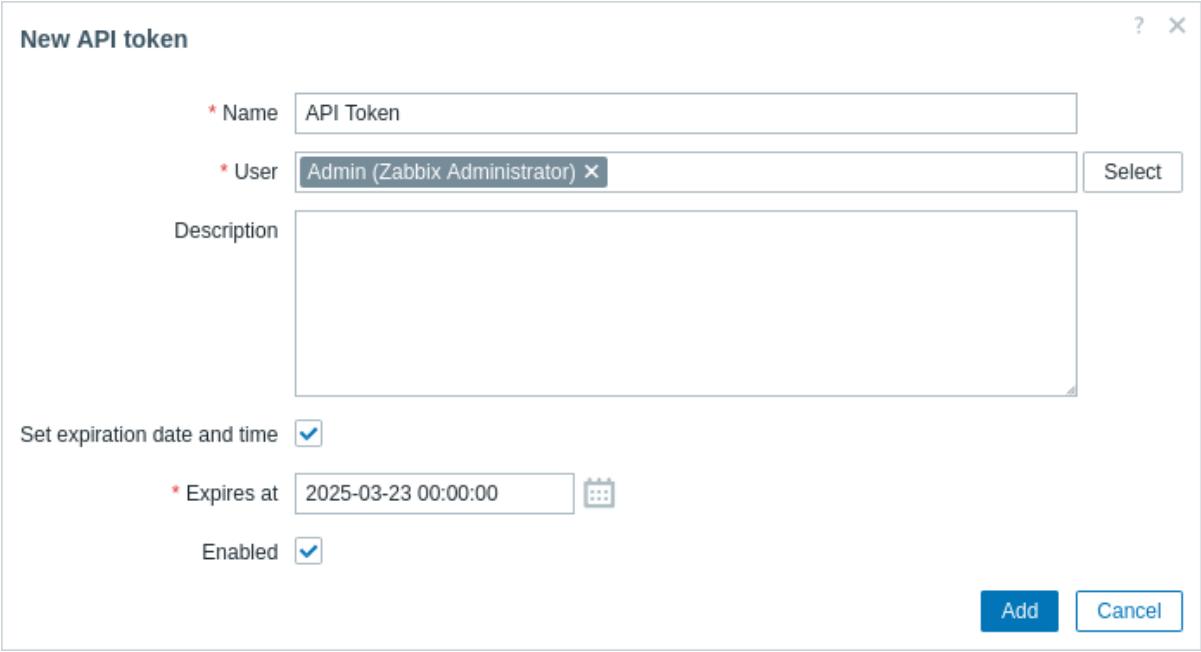
Przegląd

Ta sekcja umożliwia tworzenie tokenów API i zarządzanie nimi.
 Tokeny API służą do uwierzytelniania w API Zabbix.
 Każdy token API jest przypisany do jednego użytkownika i dziedziczy uprawnienia tego użytkownika. Żądania uwierzytelnione za pomocą tokenu API są autoryzowane zgodnie z rolą i uprawnieniami powiązanego użytkownika.



Tokeny API można filtrować według nazwy, użytkowników, do których są przypisane, daty wygaśnięcia, użytkowników, którzy utworzyli tokeny, lub statusu (włączony/wyłączony). Kliknij status tokenu na liście, aby szybko go włączyć/wyłączyć. Można również masowo włączać/wyłączać tokeny, zaznaczając je na liście, a następnie klikając przyciski *Włącz/Wyłącz* poniżej listy.

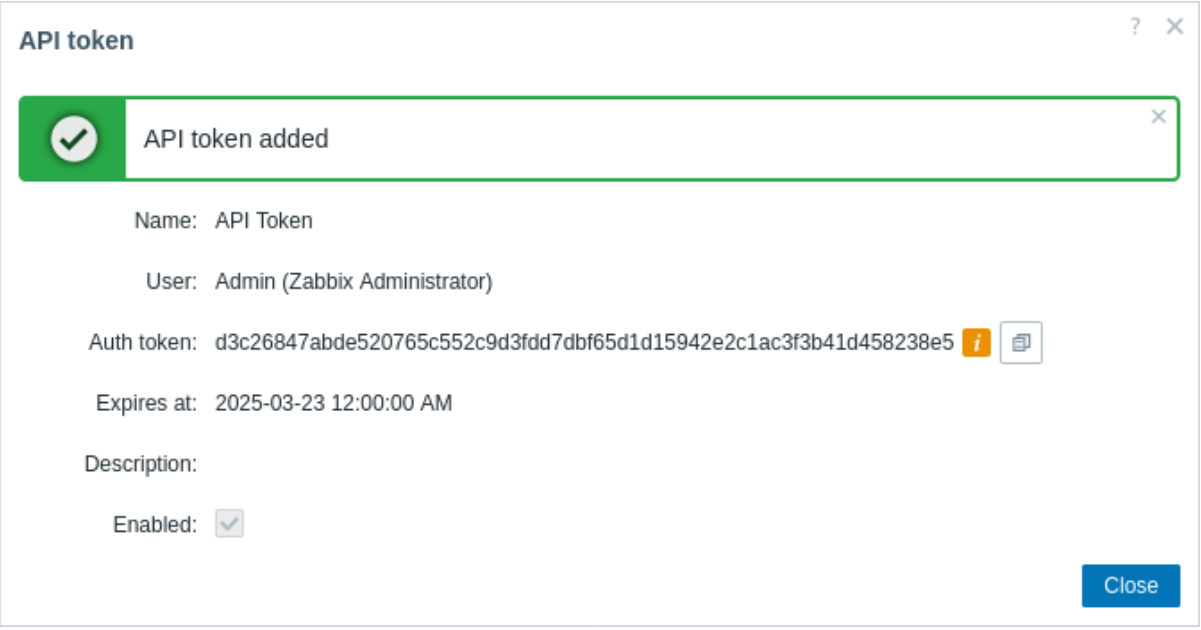
Aby utworzyć nowy token, kliknij przycisk *Utwórz token API* w prawym górnym rogu, a następnie wypełnij wymagane pola na ekranie konfiguracji tokenu:



Parametr	Opis
Nazwa	Widoczna nazwa tokenu.
Użytkownik	Użytkownik, do którego token powinien zostać przypisany. Aby szybko wybrać użytkownika, zacznij wpisywać nazwę użytkownika, imię lub nazwisko, a następnie wybierz wymaganego użytkownika z listy autouzupełniania. Możesz również kliknąć przycisk *Wybierz* i wybrać użytkownika z pełnej listy użytkowników. Token może być przypisany tylko do jednego użytkownika.
Opis	Opcjonalny opis tokenu.
Ustaw datę i godzinę wygaśnięcia	Odznacz to pole wyboru, jeśli token nie powinien mieć daty wygaśnięcia.
Data wygaśnięcia	Kliknij ikonę kalendarza, aby wybrać datę wygaśnięcia tokenu, lub wprowadź datę ręcznie w formacie YYYY-MM-DD hh:mm:ss

Parametr	Opis
Włączony	Odznacz to pole wyboru, jeśli token ma zostać utworzony w stanie wyłączonym.

Kliknij *Dodaj*, aby utworzyć token.



Skopiuj wartość *Token uwierzytelniający* i zapisz ją w bezpiecznym miejscu **przed zamknięciem strony**, a następnie kliknij *Zamknij*. Token pojawi się na liście.

Warning:
Wartości *Token uwierzytelniający* nie można wyświetlić ponownie. Jest dostępna tylko bezpośrednio po utworzeniu tokenu. Jeśli utracisz zapisany token, musisz go wygenerować ponownie, co spowoduje utworzenie nowego ciągu autoryzacyjnego.

Kliknij nazwę tokenu, aby edytować jego nazwę, opis, ustawienia daty wygaśnięcia lub status tokenu. Pamiętaj, że nie można zmienić użytkownika, do którego token jest przypisany. Kliknij przycisk *Aktualizuj*, aby zapisać zmiany. Jeśli token został utracony lub ujawniony, możesz kliknąć przycisk *Wygeneruj ponownie*, aby wygenerować nową wartość tokenu. Zostanie wyświetlone okno dialogowe z potwierdzeniem, w którym należy potwierdzić tę operację, ponieważ po jej wykonaniu wcześniej wygenerowany token stanie się nieważny.

Użytkownicy bez dostępu do sekcji menu *Administracja* mogą wyświetlać i modyfikować szczegóły tokenów przypisanych do nich w sekcji *Profil użytkownika* > *Tokeny API* section tylko wtedy, gdy w uprawnieniach ich roli użytkownika zezwolono na *Zarządzanie tokenami API*.

Uwierzytelnianie

Przegląd

Sekcja *Użytkownicy* > *Uwierzytelnianie* umożliwia określenie metody uwierzytelniania użytkowników w Zabbix oraz wewnętrznych wymagań dotyczących haseł.

Dostępne metody uwierzytelniania to uwierzytelnianie wewnętrzne, HTTP, LDAP, SAML i MFA.

Domyślne uwierzytelnianie

Domyślnie Zabbix używa **wewnętrznego** uwierzytelniania Zabbix dla wszystkich użytkowników.

Można zmienić domyślną metodę uwierzytelniania na system LDAP w całym systemie. W tym celu przejdź do karty *LDAP* i skonfiguruj parametry LDAP, a następnie wróć do karty *Uwierzytelnianie* i zmień ustawienie *Domyślne uwierzytelnianie* na LDAP.

Należy pamiętać, że metodę uwierzytelniania można dostosować na poziomie grupy użytkowników. Nawet jeśli uwierzytelnianie LDAP jest ustawione globalnie, niektóre grupy użytkowników mogą nadal być uwierzytelniane przez Zabbix. Dla tych grup opcja dostępu do frontendu musi być ustawiona na Wewnętrzny.

Można również włączyć uwierzytelnianie LDAP tylko dla określonych grup użytkowników, jeśli globalnie używane jest uwierzytelnianie wewnętrzne. W takim przypadku szczegóły uwierzytelniania LDAP można określić i stosować dla wybranych grup użytkown-

ików, dla których opcję dostępu do frontendu należy następnie ustawić na LDAP. Jeśli użytkownik należy do co najmniej jednej grupy użytkowników z uwierzytelnianiem LDAP, nie będzie mógł korzystać z wewnętrznej metody uwierzytelniania.

Metody uwierzytelniania HTTP, SAML 2.0 i MFA mogą być używane dodatkowo do domyślnej metody uwierzytelniania.

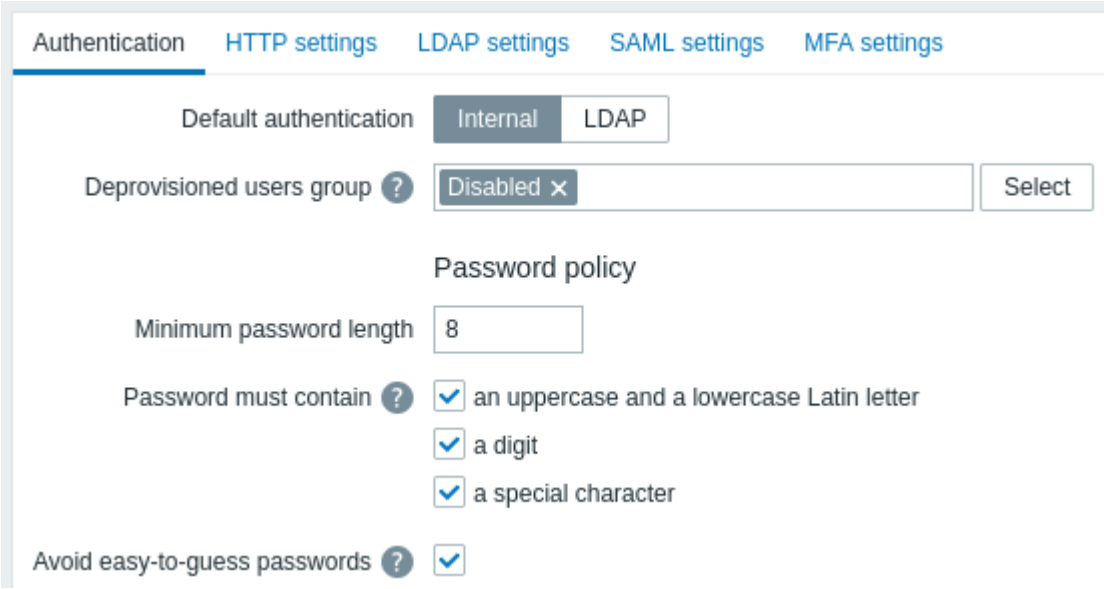
Zabbix obsługuje aprowizację just-in-time (JIT), która umożliwia tworzenie kont użytkowników w Zabbix przy pierwszym uwierzytelnieniu użytkownika zewnętrznego oraz aprowizowanie tych kont. Aprowizacja JIT jest obsługiwana dla LDAP i SAML.

Zobacz także:

- Uwierzytelnianie HTTP
- Uwierzytelnianie LDAP
- Uwierzytelnianie SAML
- Uwierzytelnianie MFA

Konfiguracja

Karta *Uwierzytelnianie* umożliwia ustawienie domyślnej metody uwierzytelniania, określenie grupy dla użytkowników wycofanych z obsługi oraz ustawienie wymagań dotyczących złożoności haseł użytkowników Zabbix.



Parametry konfiguracji:

Parametr	Opis
Domyślne uwierzytelnianie	Wybierz domyślną metodę uwierzytelniania w Zabbix - *Wewnętrzne* lub *LDAP*.
Grupa użytkowników wycofanych z obsługi	Określ grupę użytkowników dla użytkowników wycofanych z obsługi. To ustawienie jest wymagane tylko w przypadku aprowizacji JIT i dotyczy użytkowników utworzonych w Zabbix na podstawie systemów LDAP lub SAML, którzy nie muszą już być aprowizowani. Należy określić wyłączoną grupę użytkowników.
Minimalna długość hasła	Domyślnie minimalna długość hasła wynosi 8 znaków. Obsługiwany zakres: 1-70. Należy pamiętać, że hasła dłuższe niż 72 znaki zostaną obcięte.
Hasło musi zawierać	Zaznacz jedno lub kilka pól wyboru, aby wymagać użycia określonych znaków w haśle: - wielką i małą literę alfabetu łacińskiego - cyfrę - znak specjalny
Unikaj haseł łatwych do odgadnięcia	Najedź kursorem na znak zapytania, aby wyświetlić podpowiedź z listą znaków dla każdej opcji. Jeśli opcja jest zaznaczona, hasło zostanie sprawdzone pod kątem następujących wymagań: - nie może zawierać imienia, nazwiska ani nazwy użytkownika użytkownika - nie może być jednym z często używanych haseł ani haseł specyficznych dla kontekstu. Lista często używanych haseł i haseł specyficznych dla kontekstu jest generowana automatycznie na podstawie listy „Top 100k passwords” organizacji NCSC, listy „Top 1M passwords” SecLists oraz listy haseł specyficznych dla kontekstu Zabbix. Użytkownicy wewnętrzni nie będą mogli ustawić haseł znajdujących się na tej liście, ponieważ takie hasła są uznawane za słabe ze względu na ich powszechne użycie.

Zmiany wymagań dotyczących złożoności haseł nie wpłyną na istniejące hasła użytkowników, ale jeśli istniejący użytkownik zdecyduje się zmienić hasło, nowe hasło będzie musiało spełniać aktualne wymagania. Podpowiedź z listą wymagań zostanie wyświetlona obok pola *Hasło* w profilu użytkownika oraz w formularzu konfiguracji użytkownika, dostępnym z menu *Użytkownicy* > *Użytkownicy*.

HTTP

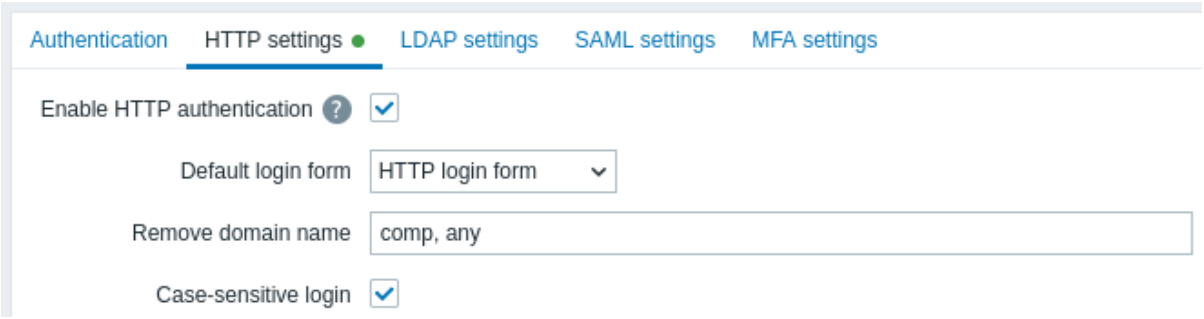
Przegląd

Uwierzytelnianie oparte na serwerze HTTP lub serwerze WWW uwierzytelnianie (np. BasicAuthentication, NTLM/Kerberos) może służyć do sprawdzania nazw użytkowników i haseł. Należy jednak pamiętać, że użytkownik musi również istnieć w Zabbix, ale jego hasło Zabbix nie będzie używane.

Attention:
Uwaga! Przed włączeniem uwierzytelniania serwera WWW upewnij się, że jest ono skonfigurowane i działa prawidłowo.

Note:
Konfigurację uwierzytelniania HTTP można wyłączyć, ustawiając `$ZBX_FEATURE_FLAGS['http_auth_enabled'] = false` w pliku konfiguracji frontendu (`zabbix.conf.php`). W takim przypadku karta *Ustawienia HTTP* z opcjami uwierzytelniania HTTP będzie ukryta.

Konfiguracja



Parametry konfiguracji:

Parametr	Opis
Włącz uwierzytelnianie HTTP	Zaznacz pole wyboru, aby włączyć uwierzytelnianie HTTP. Umieszczenie wskaźnika myszy nad wyświetli podpowiedź ostrzegającą, że w przypadku uwierzytelniania przez serwer WWW wszyscy użytkownicy (nawet ci, dla których dostęp do frontendu jest ustawiony na LDAP/Wewnętrzne) będą uwierzytelniani przez serwer WWW, a nie przez Zabbix.
Domyślny formularz logowania	Określ, czy nieuwierzytelnieni użytkownicy mają być kierowani do: **Formularz logowania Zabbix** - standardowa strona logowania Zabbix. **Formularz logowania HTTP** - strona logowania HTTP. Zaleca się włączenie uwierzytelniania opartego na serwerze WWW tylko dla strony `index_http.php`. Jeśli dla opcji *Domyślny formularz logowania* ustawiono wartość „Strona logowania HTTP”, użytkownik zostanie zalogowany automatycznie, jeśli moduł uwierzytelniania serwera WWW ustawi prawidłowy login użytkownika w zmiennej `$_SERVER`. Obsługiwane klucze `$_SERVER` to `PHP_AUTH_USER`, `REMOTE_USER`, `AUTH_USER`.
Usuń nazwę domeny	Rozdzielana przecinkami lista nazw domen, które powinny zostać usunięte z nazwy użytkownika. Na przykład `comp,any` - jeśli nazwa użytkownika to „Admin@any” lub „comp\Admin”, użytkownik zostanie zalogowany jako „Admin”; jeśli nazwa użytkownika to „notacompany\Admin”, logowanie zostanie odrzucone.
Logowanie z uwzględnieniem wielkości liter	Usuń zaznaczenie pola wyboru, aby wyłączyć uwzględnianie wielkości liter podczas logowania przy użyciu nazw użytkowników (domyślnie włączone). Wyłączenie uwzględniania wielkości liter podczas logowania umożliwia na przykład zalogowanie się jako „admin”, nawet jeśli użytkownik Zabbix ma nazwę „Admin” lub „ADMIN”. Należy pamiętać, że jeśli uwzględnianie wielkości liter podczas logowania jest wyłączone i istnieje wielu użytkowników Zabbix o podobnych nazwach (np. Admin i admin), logowanie tych użytkowników będzie zawsze odrzucane z następującym komunikatem o błędzie: „Uwierzytelnianie nie powiodło się: podane dane uwierzytelniające nie są unikatowe.”

Note:
W przypadku użytkowników wewnętrznych, którzy nie mogą zalogować się przy użyciu danych uwierzytelniających HTTP (gdy jako domyślny ustawiono formularz logowania HTTP), co prowadzi do błędu 401, można dodać wiersz `ErrorDocument 401 /index.php?form=default` do dyrektyw uwierzytelniania podstawowego. Spowoduje to przekierowanie do standardowego formularza logowania Zabbix.

LDAP

Przegląd

Zewnętrzne uwierzytelnianie LDAP może być używane do sprawdzania nazw użytkowników i haseł.

Uwierzytelnianie LDAP w Zabbix działa co najmniej z Microsoft Active Directory i OpenLDAP.

Jeśli skonfigurowano tylko logowanie LDAP, użytkownik musi również istnieć w Zabbix, jednak jego hasło Zabbix nie będzie używane. Jeśli uwierzytelnianie zakończy się powodzeniem, Zabbix dopasuje lokalną nazwę użytkownika do atrybutu nazwy użytkownika zwróconego przez LDAP.

Provisioning użytkowników

Można skonfigurować JIT (just-in-time) **provisioning użytkowników** LDAP. W takim przypadku użytkownik nie musi już istnieć w Zabbix. Konto użytkownika może zostać utworzone, gdy użytkownik loguje się do Zabbix po raz pierwszy.

Gdy użytkownik LDAP wprowadzi swój login i hasło LDAP, Zabbix sprawdza, czy ten użytkownik istnieje na *domyślnym* serwerze LDAP. Jeśli użytkownik istnieje i nie ma jeszcze konta w Zabbix, w Zabbix tworzony jest nowy użytkownik, który może się zalogować.

Użytkownik utworzony za pomocą JIT provisioning jest powiązany z serwerem LDAP (katalogiem), który w momencie utworzenia jest ustawiony jako domyślny. Późniejsza zmiana domyślnego serwera LDAP nie zmienia ani nie aktualizuje serwera LDAP powiązanego z użytkownikami, którzy zostali już utworzeni za pomocą provisioning.

Attention:
Jeśli JIT provisioning jest włączony, na karcie *Authentication* należy określić grupę użytkowników przeznaczoną dla użytkowników usuniętych z provisioningu.

JIT provisioning umożliwia również aktualizowanie utworzonych kont użytkowników na podstawie zmian w LDAP. Na przykład jeśli użytkownik zostanie przeniesiony z jednej grupy LDAP do innej, zostanie również przeniesiony z jednej grupy do innej w Zabbix; jeśli użytkownik zostanie usunięty z grupy LDAP, zostanie również usunięty z grupy w Zabbix, a jeśli nie należy do żadnej innej grupy, zostanie dodany do grupy użytkowników przeznaczonej dla użytkowników usuniętych z provisioningu. Utworzone konta użytkowników są aktualizowane zgodnie ze skonfigurowanym okresem provisioningu lub podczas logowania użytkownika do Zabbix.

Należy pamiętać, że provisioning w tle jest wykonywany przez frontend Zabbix, gdy użytkownik korzysta z niego lub ma przynajmniej otwartą stronę frontendu w przeglądarce. Nie ma dedykowanych procesów działających w tle do provisioningu użytkowników.

LDAP obsługuje trzy sposoby wiązania z katalogiem na potrzeby uwierzytelniania i wyszukiwania:

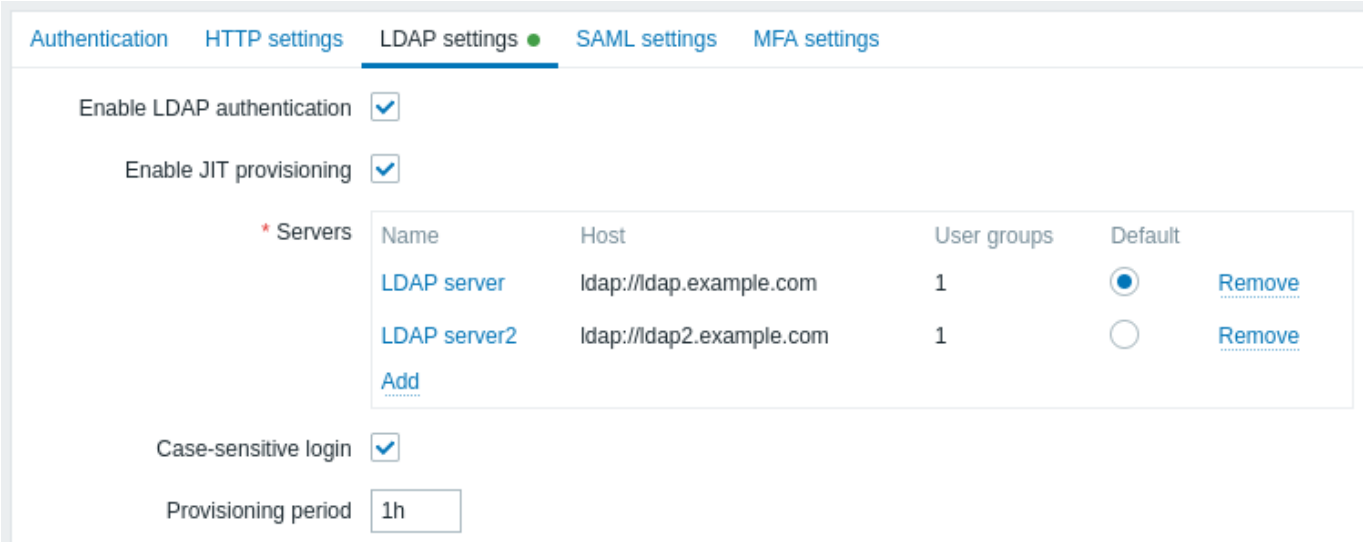
- Wiązanie anonimowe - nie podano *Bind DN* / *Bind password*, a serwer LDAP zezwala na zapytania anonimowe.
- Dedykowany użytkownik do wiązania (konto usługi) - określone konto LDAP jest ustawione w polach *Bind DN* / *Bind password* i używane przez Zabbix do wyszukiwania oraz provisioningu. Jest to zalecana i najbardziej elastyczna opcja, ponieważ Zabbix może wykonywać wyszukiwanie i provisioning w tle bez poświadczeń użytkownika końcowego.
- Bezpośrednie wiązanie użytkownika - Zabbix wykonuje wiązanie przy użyciu poświadczeń wprowadzonych przez użytkownika podczas logowania (nie skonfigurowano *Bind DN* / *Bind password*); jest to konfigurowane przez uwzględnienie symbolu zastępczego, takiego jak `uid=%{user}`, w Base DN. W tym trybie Zabbix ma dostęp do hasła użytkownika tylko podczas interaktywnego logowania. W rezultacie działania provisioningu, które wymagają uwierzytelnienia w LDAP poza sesją logowania użytkownika (na przykład za pomocą przycisku *Provision now* lub podczas uruchamiania provisioningu w tle, gdy użytkownik nie jest aktywnie zalogowany), nie mogą zostać uwierzytelnione i dlatego nie będą działać. Provisioning i aktualizacje w przypadku bezpośredniego wiązania użytkownika odbywają się wyłącznie w momencie logowania użytkownika.

Wiele serwerów

W razie potrzeby można zdefiniować kilka serwerów LDAP. Na przykład do uwierzytelniania innej grupy użytkowników można użyć innego serwera. Po skonfigurowaniu serwerów LDAP w konfiguracji grupy użytkowników będzie można wybrać wymagany serwer LDAP dla odpowiedniej grupy użytkowników.

Jeśli użytkownik należy do wielu grup użytkowników i wielu serwerów LDAP, do uwierzytelniania zostanie użyty pierwszy serwer z listy serwerów LDAP posortowanej według nazwy w porządku rosnącym.

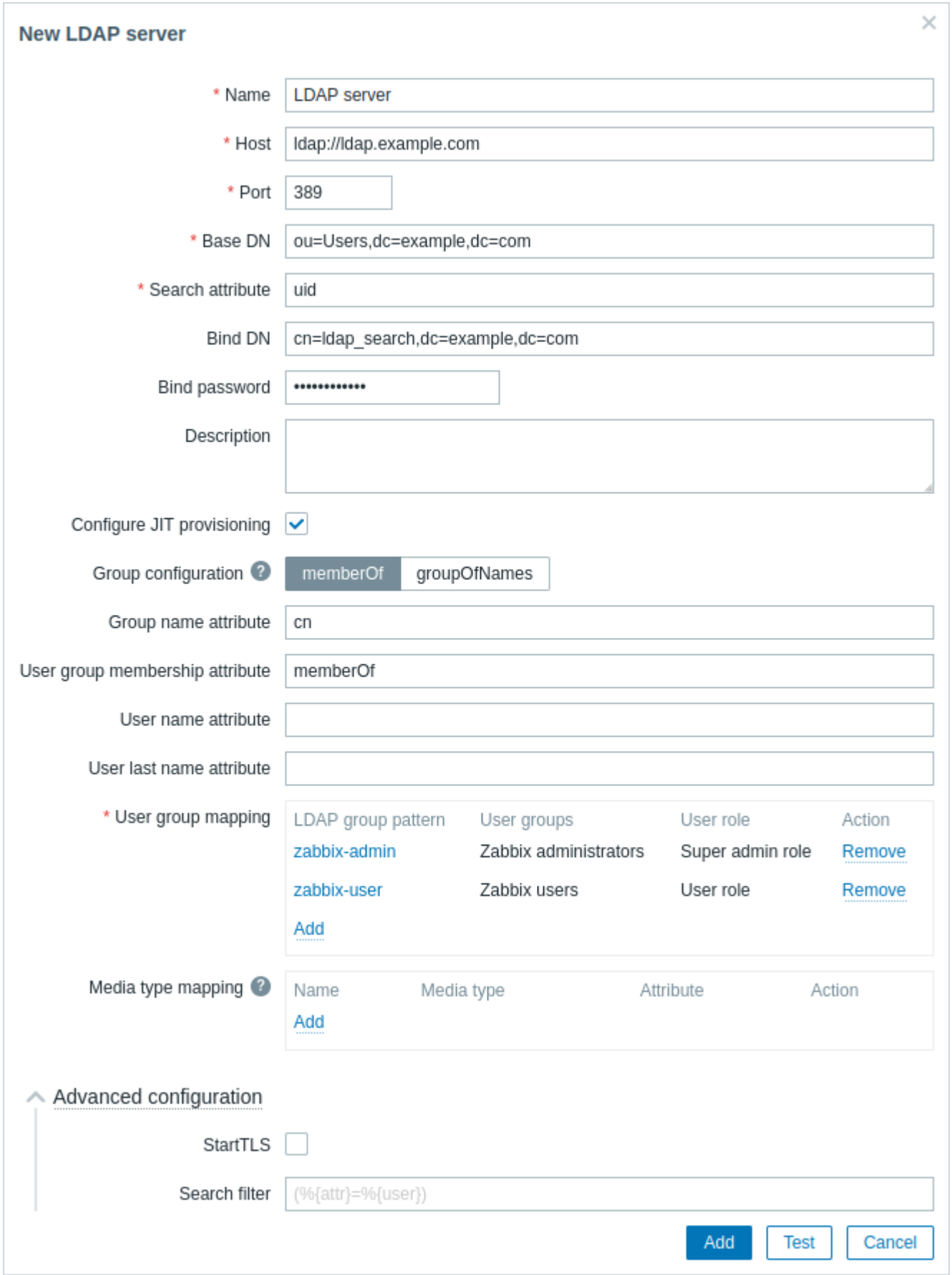
Konfiguracja



Parametry konfiguracji:

Parametr	Opis
Włącz uwierzytelnianie LDAP	Zaznacz pole wyboru, aby włączyć uwierzytelnianie LDAP.
Włącz aprowizację JIT	Zaznacz pole wyboru, aby włączyć aprowizację JIT.
Serwery	Kliknij *Dodaj*, aby skonfigurować serwer LDAP (zobacz poniżej Konfiguracja serwera LDAP).
Logowanie z uwzględnieniem wielkości liter	Odznacz pole wyboru, aby wyłączyć uwzględnianie wielkości liter podczas logowania przy użyciu nazw użytkowników (domyślnie włączone). Wyłączenie uwzględniania wielkości liter podczas logowania umożliwia na przykład zalogowanie się jako „admin”, nawet jeśli użytkownik Zabbix ma nazwę „Admin” lub „ADMIN”. Należy pamiętać, że jeśli uwzględnianie wielkości liter podczas logowania jest wyłączone i istnieje wielu użytkowników Zabbix o podobnych nazwach użytkowników (np. Admin i admin), logowanie tych użytkowników będzie zawsze odrzucane z następującym komunikatem o błędzie: „Authentication failed: supplied credentials are not unique.”
Okres aprowizacji	Ustaw okres aprowizacji, czyli częstotliwość aprowizowania zalogowanego użytkownika podczas pracy z frontendem.

Konfiguracja serwera LDAP



Parametry konfiguracji serwera LDAP:

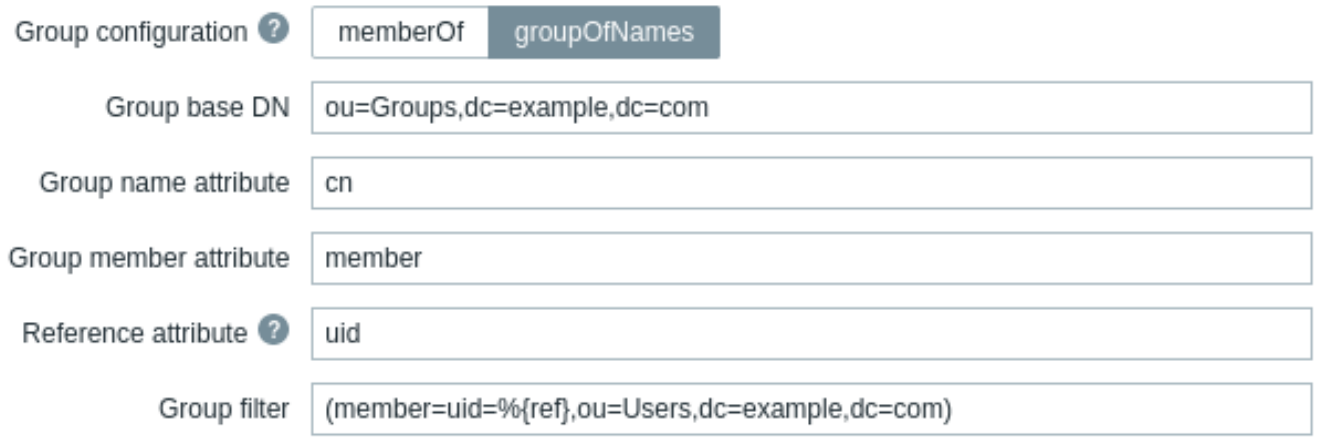
Parameter	Description
Name	Nazwa serwera LDAP w konfiguracji Zabbix.

Parameter	Description
Host	Nazwa hosta, adres IP lub URI serwera LDAP. Przykłady: ldap.example.com, 127.0.0.1, ldap://ldap.example.com W przypadku bezpiecznego serwera LDAP użyj protokołu *ldaps* i nazwy hosta. Przykład: ldaps://ldap.example.com W przypadku OpenLDAP 2.x.x i nowszych można użyć pełnego URI LDAP w postaci ldap://hostname:port lub ldaps://hostname:port.
Port	Port serwera LDAP. Domyślnie 389. W przypadku bezpiecznego połączenia LDAP numer portu zwykle wynosi 636. Nie jest używany przy korzystaniu z pełnych URI LDAP.
Base DN	Bazowa ścieżka do kont użytkowników na serwerze LDAP: ou=Users,ou=system (dla OpenLDAP), DC=company,DC=com (dla Microsoft Active Directory) uid=%{user},dc=example,dc=com (dla bezpośredniego wiązania użytkownika, zobacz uwagę poniżej)
Search attribute	Atrybut konta LDAP używany do wyszukiwania: uid (dla OpenLDAP), sAMAccountName (dla Microsoft Active Directory)
Bind DN	Konto LDAP do wiązania i wyszukiwania na serwerze LDAP, przykłady: uid=ldap_search,ou=system (dla OpenLDAP), CN=ldap_search,OU=user_group,DC=company,DC=com (dla Microsoft Active Directory) Obsługiwane jest również anonimowe wiązanie. Należy pamiętać, że anonimowe wiązanie potencjalnie udostępnia konfigurację domeny nieautoryzowanym użytkownikom (informacje o użytkownikach, komputerach, serwerach, grupach, usługach itp.). Ze względów bezpieczeństwa wyłącz anonimowe wiązania na hostach LDAP i zamiast tego używaj uwierzytelnionego dostępu.
Bind password	Hasło LDAP konta używanego do wiązania i wyszukiwania na serwerze LDAP.
Description	Opis serwera LDAP.
Configure JIT provisioning	Zaznacz to pole wyboru, aby wyświetlić opcje związane z JIT provisioning.
Group configuration	Wybierz metodę konfiguracji grup: **memberOf** - poprzez wyszukiwanie użytkowników i ich atrybutu członkostwa w grupach **groupOfNames** - poprzez wyszukiwanie grup za pomocą atrybutu member Należy pamiętać, że memberOf jest preferowane, ponieważ działa szybciej; użyj groupOfNames, jeśli serwer LDAP nie obsługuje `memberOf` lub wymagane jest filtrowanie grup.
Group name attribute	Określ atrybut, z którego ma być pobierana nazwa grupy ze wszystkich obiektów w atrybucie `memberOf` (zobacz pole *User group membership attribute*) Nazwa grupy jest niezbędna do mapowania grup użytkowników.
User group membership attribute	Określ atrybut zawierający informacje o grupach, do których należy użytkownik (np. `memberOf`). Na przykład atrybut memberOf może zawierać takie informacje: `memberOf=cn=zabbix-admin,ou=Groups,dc=example,dc=com` To pole jest dostępne tylko dla metody memberOf.
User name attribute	Określ atrybut zawierający imię użytkownika.
User last name attribute	Określ atrybut zawierający nazwisko użytkownika.
User group mapping	Mapuj wzorzec grupy użytkowników LDAP na grupę użytkowników Zabbix i rolę użytkownika. Jest to wymagane do określenia, jaką grupę użytkowników/rolę otrzyma aprowizowany użytkownik w Zabbix. Kliknij *Add*, aby dodać mapowanie. Pole *LDAP group pattern* obsługuje symbole wieloznaczne. Nazwa grupy musi odpowiadać istniejącej grupie. Jeśli użytkownik LDAP pasuje do kilku grup użytkowników Zabbix, stanie się członkiem wszystkich z nich. Jeśli użytkownik pasuje do kilku ról użytkowników Zabbix, otrzyma tę o najwyższym poziomie uprawnień spośród nich.
Media type mapping	Mapuj atrybuty LDAP użytkownika media (np. e-mail) na media użytkownika Zabbix do wysyłania powiadomień (wartość atrybutu jest używana jako pole media *Send to*).
Advanced configuration	Kliknij nagłówek *Advanced configuration*, aby wyświetlić zaawansowane opcje konfiguracji (patrz poniżej).
StartTLS	Zaznacz pole wyboru, aby używać operacji StartTLS podczas łączenia z serwerem LDAP. Połączenie zostanie przerwane, jeśli serwer nie obsługuje StartTLS. StartTLS nie może być używany z serwerami korzystającymi z protokołu *ldaps*.

Parameter	Description
Search filter	Zdefiniuj własny ciąg podczas uwierzytelniania użytkownika w LDAP. Obsługiwane są następujące symbole zastępcze: `%{attr}` - nazwa atrybutu wyszukiwania (uid, sAMAccountName) `%{user}` - wartość nazwy użytkownika do uwierzytelnienia Na przykład, aby przeprowadzić wyszukiwanie z uwzględnieniem wielkości liter w środowisku LDAP lub Microsoft Active Directory, które domyślnie nie rozróżnia wielkości liter, ciąg można zdefiniować następująco: `(%{attr}:caseExactMatch:=%{user})`. Jeśli filtr nie zostanie dostosowany, LDAP użyje domyślnego: `(%{attr}=%{user})`.

Note:
Aby skonfigurować serwer LDAP dla **bezpośredniego wiązania użytkownika**, dodaj atrybut uid=%{user} do parametru *Base DN* (na przykład *uid=%{user},dc=example,dc=com*) i pozostaw parametry *BindDN* oraz *Bind password* puste. Podczas uwierzytelniania symbol zastępczy %{user} zostanie zastąpiony nazwą użytkownika wprowadzoną podczas logowania.
 W przypadku bezpośredniego wiązania użytkownika Zabbix ma dostęp do poświadczeń użytkownika tylko podczas interaktywnego logowania. Dlatego zadania aprowizacji działające bez interaktywnego logowania (na przykład przycisk *Provision now* lub metoda API `user.provision`) zignorują użytkowników uwierzytelniających się za pomocą bezpośredniego wiązania użytkownika, ponieważ Zabbix nie może wykonać wiązania w ich imieniu. Użyj anonimowego wiązania lub dedykowanego użytkownika wiążącego (konta usługowego), jeśli potrzebujesz, aby aprowizacja i wyszukiwania LDAP działały z frontend bez konieczności logowania się użytkownika końcowego.

Następujące pola są specyficzne dla metody *Group configuration* "groupOfNames":



Parameter	Description
Group base DN	Bazowa ścieżka do grup na serwerze LDAP.
Group name attribute	Określ atrybut, z którego ma być pobierana nazwa grupy w określonej bazowej ścieżce do grup. Nazwa grupy jest niezbędna do mapowania grup użytkowników.
Group member attribute	Określ atrybut zawierający informacje o członkach grupy w LDAP (np. `member`).
Reference attribute	Określ atrybut referencyjny dla filtra grupy (zobacz pole *Group filter*). Następnie użyj `%{ref}` w filtrze grupy, aby pobrać wartości dla atrybutu określonego tutaj.
Group filter	Określ filtr do pobrania grupy, której użytkownik jest członkiem. Na przykład `(member=uid=%{ref},ou=Users,dc=example,dc=com)` dopasuje "User1", jeśli atrybut member grupy ma wartość `uid=User1,ou=Users,dc=example,dc=com`, i zwróci grupę, której członkiem jest "User1".

Warning:
W przypadku problemów z certyfikatami, aby połączenie z bezpiecznym LDAP (ldaps) działało, może być konieczne dodanie linii `TLS_REQCERT allow` do pliku konfiguracyjnego `/etc/openldap/ldap.conf`. Może to obniżyć bezpieczeństwo połączenia z katalogiem LDAP.

Note:
Zaleca się utworzenie oddzielnego konta LDAP (*Bind DN*) do wykonywania wiązania i wyszukiwania na serwerze LDAP z minimalnymi uprawnieniami w LDAP, zamiast używania rzeczywistych kont użytkowników (używanych do logowania do frontend Zabbix).
 Takie podejście zapewnia większe bezpieczeństwo i nie wymaga zmiany *Bind password*, gdy użytkownik zmienia własne hasło na serwerze LDAP.
 W powyższej tabeli jest to nazwa konta *ldap_search*.

Testowanie dostępu

Przycisk *Testuj* umożliwia przetestowanie dostępu użytkownika:

Parametr	Opis
Login	Nazwa użytkownika LDAP do przetestowania (wstępnie wypełniona bieżącą nazwą użytkownika z frontend Zabbix). Ta nazwa użytkownika musi istnieć na serwerze LDAP. Zabbix nie aktywuje uwierzytelniania LDAP, jeśli nie będzie w stanie uwierzytelnić testowanego użytkownika.
Hasło użytkownika	Hasło użytkownika LDAP do przetestowania.

SAML

Przegląd

SAML 2.0 uwierzytelnianie może być używane do logowania się do Zabbix.

Jeśli logowanie SAML jest skonfigurowane bez aprowizacji użytkowników JIT, użytkownik musi już istnieć w Zabbix; hasło Zabbix użytkownika nie jest używane do uwierzytelniania. Po pomyślnym uwierzytelnieniu SAML Zabbix dopasowuje atrybut nazwy użytkownika otrzymany z SAML do istniejącej lokalnej nazwy użytkownika Zabbix.

Provisioning użytkowników

Można skonfigurować **provisioning użytkowników** JIT (just-in-time) dla użytkowników SAML. W takim przypadku nie jest wymagane, aby użytkownik istniał już w Zabbix. Konto użytkownika może zostać utworzone, gdy użytkownik zaloguje się do Zabbix po raz pierwszy.

Attention:
Jeśli provisioning JIT jest włączony, na karcie *Authentication* należy określić grupę użytkowników dla użytkowników, którym odebrano provisioning.

Oprócz provisioningu JIT można również włączyć i skonfigurować provisioning SCIM (System for Cross-domain Identity Management) — *ciągłe* zarządzanie kontami użytkowników dla tych użytkowników, którzy zostali utworzeni przez provisioning użytkowników. Provisioning SCIM wymaga tokena API Zabbix (z uprawnieniami Super admin) do uwierzytelniania w Zabbix.

Na przykład, jeśli użytkownik zostanie przeniesiony z jednej grupy SAML do innej, użytkownik zostanie również przeniesiony z jednej grupy do innej w Zabbix; jeśli użytkownik zostanie usunięty z grupy SAML, użytkownik zostanie również usunięty z grupy w Zabbix i, jeśli nie należy do żadnej innej grupy, zostanie dodany do grupy użytkowników dla użytkowników, którym odebrano provisioning.

Jeśli SCIM jest włączony i skonfigurowany, użytkownik SAML zostanie provisionowany w momencie zalogowania się do Zabbix i będzie stale aktualizowany na podstawie zmian w SAML. Istniejący już użytkownicy SAML nie będą provisionowani, a aktualizowani będą tylko użytkownicy provisionowani. Należy pamiętać, że do użytkownika zostaną dodane tylko prawidłowe media, gdy użytkownik zostanie provisionowany lub zaktualizowany.

Jeśli SCIM nie jest włączony, użytkownik SAML zostanie provisionowany (a później zaktualizowany) w momencie zalogowania się do Zabbix.

Note:
Jeśli uwierzytelnianie SAML jest włączone, użytkownicy będą mogli wybrać logowanie lokalne lub przez logowanie jednokrotne SAML. Jeśli używany jest provisioning JIT, możliwe jest tylko logowanie jednokrotne.

Konfigurowanie dostawcy tożsamości

Aby pracować z Zabbix, dostawca tożsamości SAML (onelogin.com, auth0.com, okta.com itp.) musi zostać skonfigurowany w następujący sposób:

- *Assertion Consumer URL* należy ustawić na `<path_to_zabbix_ui>/index_sso.php?acs`

- *Single Logout URL* należy ustawić na `<path_to_zabbix_ui>/index_sso.php?sls`

Przykłady `<path_to_zabbix_ui>`: `https://example.com/zabbix/ui`, `http://another.example.com/zabbix`, `http://<any_public_ip_address>/zabbix`

Konfigurowanie Zabbix

Attention:
Jeśli chcesz używać uwierzytelniania SAML w frontendzie, musisz zainstalować php-openssl.

Aby używać uwierzytelniania SAML, należy skonfigurować Zabbix w następujący sposób:

1. Udostępnij klucz prywatny SP i certyfikaty. Aby aktywować SAML za pośrednictwem frontendu, należy udostępnić certyfikat IdP (wkleić go lub wybrać plik). Jeśli wybrano opcję *Podpisz* lub *Szyfruj*, należy udostępnić certyfikat SP i klucz prywatny SP.

Jeśli w pliku konfiguracyjnym frontendu (`zabbix.conf.php`) ustawiono `$SSO['CERT_STORAGE'] = 'database'`, podczas konfigurowania SAML można wkleić tekst certyfikatu lub przesłać plik certyfikatu w frontendzie — pliki w systemie plików nie są wymagane.

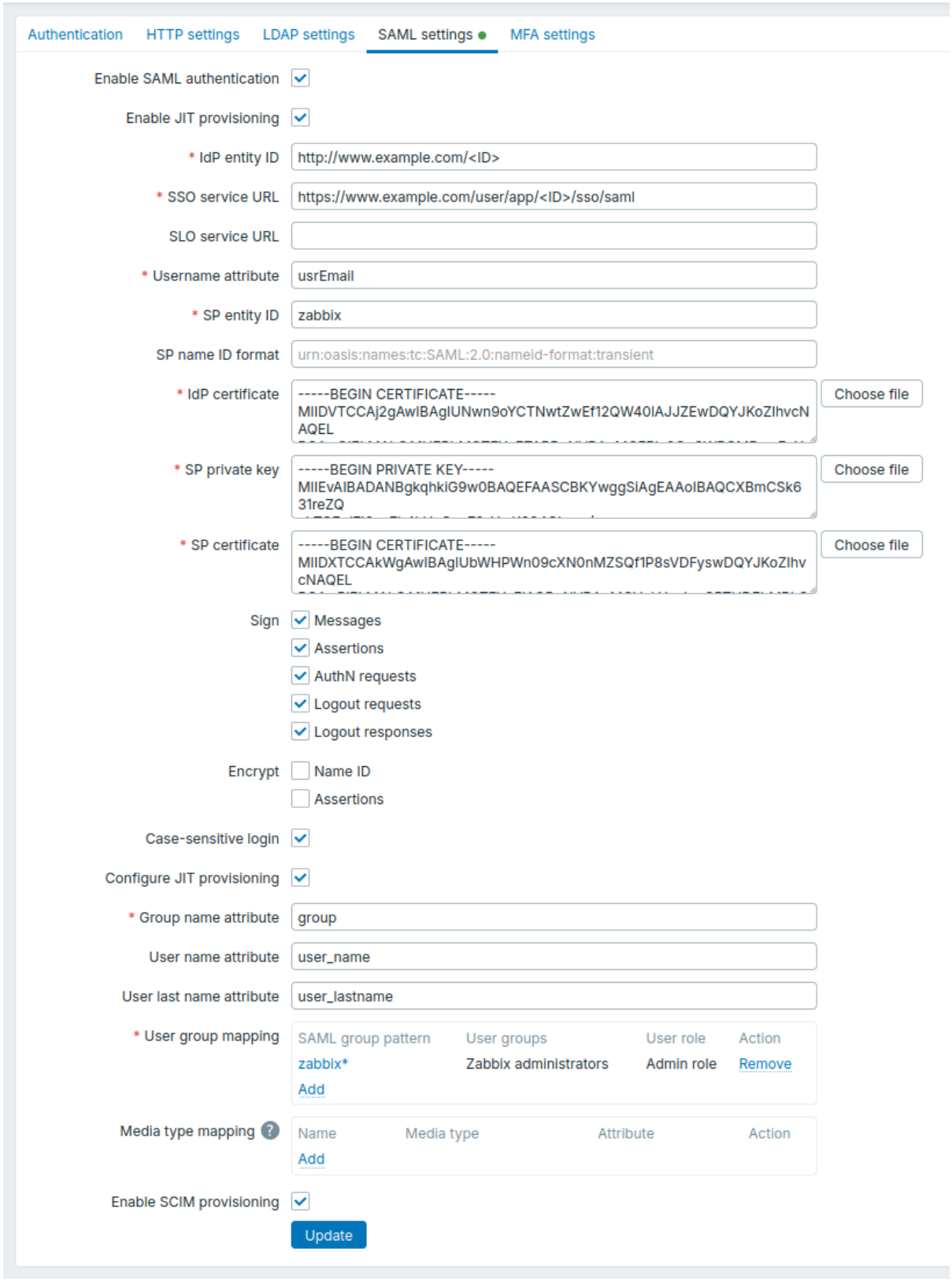
Jeśli w pliku konfiguracyjnym ustawiono `$SSO['CERT_STORAGE'] = 'file'`, certyfikat musi być dostępny w systemie plików (domyślnie w `ui/conf/certs` lub w ścieżce skonfigurowanej w pliku konfiguracyjnym), a frontend nie będzie przechowywać certyfikatów w bazie danych. Jeśli `$SSO['CERT_STORAGE']` nie jest ustawione lub jest zakomentowane, przyjmuje się przechowywanie w plikach.

Domyślnie Zabbix będzie szukać plików w następujących lokalizacjach:

- ui/conf/certs/sp.key - plik klucza prywatnego SP
- ui/conf/certs/sp.crt - plik certyfikatu SP
- ui/conf/certs/idp.crt - plik certyfikatu IdP

Attention:
Materiały certyfikatów można importować za pośrednictwem frontendu, wklejając tekst lub przesyłając pliki. Chociaż w niektórych środowiskach dla wygody dozwolone są importy niezaszyfrowane, zdecydowanie zaleca się korzystanie z szyfrowanego transportu (na przykład włączenie HTTPS/TLS dla frontendu Zabbix). Jeśli certyfikaty/klucze SAML są przechowywane w bazie danych, włącz TLS/SSL dla połączeń z bazą danych, aby chronić ruch między aplikacją a bazą danych.
 Zaszyfruj kopie zapasowe baz danych, które mogą zawierać certyfikaty SAML lub klucze prywatne, oraz ogranicz dostęp do plików kopii zapasowych i tabel bazy danych przechowujących materiały SAML. Ogranicz uprawnienia użytkownika bazy danych do niezbędnego minimum.

2. Wszystkie najważniejsze ustawienia można skonfigurować w frontendzie Zabbix. Możliwe jest jednak określenie dodatkowych ustawień w pliku konfiguracyjnym.



Parametry konfiguracji dostępne w frontendzie Zabbix:

Parametr	Opis
Włącz uwierzytelnianie SAML	Zaznacz pole wyboru, aby włączyć uwierzytelnianie SAML.
Włącz aprowizację JIT	Zaznacz pole wyboru, aby włączyć aprowizację użytkowników JIT.
Identyfikator encji IDP	Unikatowy identyfikator encji w ramach dostawcy tożsamości SAML.
URL usługi SSO	Adres URL, na który użytkownicy zostaną przekierowani podczas logowania.
URL usługi SLO	Adres URL, na który użytkownicy zostaną przekierowani podczas wylogowywania. Jeśli pozostawisz to pole puste, usługa SLO nie będzie używana.

Parametr	Opis
Atrybut nazwy użytkownika	Atrybut SAML używany jako nazwa użytkownika podczas logowania do Zabbix. Lista obsługiwanych wartości jest określana przez dostawcę tożsamości. Przykłady: 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
Identyfikator encji SP	Unikatowy identyfikator dostawcy usług (jeśli nie pasuje, operacja zostanie odrzucona). Można określić adres URL lub dowolny ciąg danych.
Format identyfikatora nazwy SP	Żądanie określonego formatu identyfikatora nazwy w odpowiedzi. Przykłady: 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
Certyfikat IdP	Zawartość certyfikatu dostawcy usług (SP) na potrzeby konfiguracji usługi jednokrotnego logowania (SSO) SAML.
Klucz prywatny SP	Zawartość klucza prywatnego dostawcy usług (SP) na potrzeby konfiguracji usługi jednokrotnego logowania (SSO) SAML. Zapewnia bezpieczne uwierzytelnianie i wymianę danych z dostawcą tożsamości (IdP).
Certyfikat SP	Zawartość certyfikatu dostawcy usług (SP) na potrzeby konfiguracji usługi jednokrotnego logowania (SSO) SAML.
Podpisz	Usuń zaznaczenie pól wyboru, aby wyłączyć podpisy SAML dla odpowiednich encji: *Wiadomości* *Asercje* *Żądania AuthN* *Żądania wylogowania* *Odpowiedzi wylogowania*
Szyfruj	Zaznacz pola wyboru, aby wybrać encje, dla których ma być włączone szyfrowanie SAML: *Identyfikator nazwy* *Asercje*
Logowanie z uwzględnieniem wielkości liter	Usuń zaznaczenie pola wyboru, aby wyłączyć rozróżnianie wielkości liter podczas logowania przy użyciu nazw użytkowników (domyślnie włączone). Wyłączenie rozróżniania wielkości liter podczas logowania umożliwia na przykład zalogowanie się jako „admin”, nawet jeśli użytkownik Zabbix ma nazwę „Admin” lub „ADMIN”. Należy pamiętać, że jeśli rozróżnianie wielkości liter podczas logowania jest wyłączone i istnieje wielu użytkowników Zabbix o podobnych nazwach (np. Admin i admin), logowanie tych użytkowników będzie zawsze odrzucane z następującym komunikatem o błędzie: „Uwierzytelnianie nie powiodło się: podane dane uwierzytelniające nie są unikatowe”.
Konfiguruj aprowizację JIT	Zaznacz to pole wyboru, aby wyświetlić opcje związane z aprowizacją użytkowników JIT.
Atrybut nazwy grupy	Określ atrybut nazwy grupy na potrzeby aprowizacji użytkowników JIT.
Atrybut nazwy użytkownika	Określ atrybut nazwy użytkownika na potrzeby aprowizacji użytkowników JIT.
Atrybut nazwiska użytkownika	Określ atrybut nazwiska użytkownika na potrzeby aprowizacji użytkowników JIT.
Mapowanie grup użytkowników	Zmapuj wzorzec grupy użytkownika SAML na grupę użytkowników Zabbix i rolę użytkownika. Jest to wymagane do określenia grupy/roli użytkownika, którą otrzyma aprowizowany użytkownik w Zabbix. Kliknij *Dodaj*, aby dodać mapowanie. Pole *Wzorzec grupy SAML* obsługuje symbole wieloznaczne. Nazwa grupy musi odpowiadać istniejącej grupie. Jeśli użytkownik SAML pasuje do kilku grup użytkowników Zabbix, stanie się członkiem wszystkich tych grup. Jeśli użytkownik pasuje do kilku ról użytkowników Zabbix, otrzyma najwyższy poziom uprawnień spośród nich.

Parametr	Opis
Mapowanie typów mediów	Zmapuj atrybuty mediów SAML użytkownika (np. adres e-mail) na media użytkownika Zabbix w celu wysyłania powiadomień.
Włącz aprowizację SCIM	Zaznacz to pole wyboru, aby włączyć aprowizację SCIM 2.0.

Przykłady konfigurowania dostawców tożsamości SAML na potrzeby logowania i aprowizacji użytkowników w Zabbix znajdziesz tutaj:

- Microsoft Azure AD
- Okta
- Onelogin

Uwagi dotyczące aprowizacji SCIM

W przypadku aprowizacji SCIM określ ścieżkę do frontend Zabbix i dodaj do niej `api_scim.php` po stronie dostawcy tożsamości, tj.:

`https://<path-to-zabbix-ui>/api_scim.php`

Atrybuty użytkownika używane w Zabbix (nazwa użytkownika, imię użytkownika, nazwisko użytkownika i atrybuty mediów) należy dodać jako atrybuty niestandardowe. W razie potrzeby zewnętrzna przestrzeń nazw powinna być taka sama jak schemat użytkownika: `urn:ietf:params:scim:schemas:core:2.0:User`.

Ustawienia zaawansowane

Dodatkowe parametry SAML można skonfigurować w pliku konfiguracyjnym frontend Zabbix (`zabbix.conf.php`):

- `$SSO['SP_KEY']`
- `$SSO['SP_CERT']`
- `$SSO['IDP_CERT']`
- `$SSO['SETTINGS']`

Przykłady konfiguracji

Przykład 1: Łączenie się z instancją Zabbix za pośrednictwem load balancera z użyciem TLS/SSL

Jeśli do łączenia się z instancją Zabbix używany jest load balancer, który korzysta z TLS/SSL, podczas gdy Zabbix nie korzysta z tego protokołu, określ parametry `baseurl`, `strict` i `use_proxy_headers` w następujący sposób:

```
$SSO['SETTINGS'] = [
    'strict' => false,
    'baseurl' => 'https://zabbix.example.com/zabbix/',
    'use_proxy_headers' => true
];
```

Przykład 2: Konfigurowanie niestandardowych algorytmów podpisu i skrótu 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',
        // ...
    ],
    // ...
];
```

Przykład 3: Konfigurowanie SSO z uwierzytelnianiem Kerberos za pomocą ADFS

```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => [
            'urn:oasis:names:tc:SAML:2.0:ac:classes:Kerberos',
        ],
        'requestedAuthnContextComparison' => 'exact'
    ]
];
```

W tym przypadku w polu *SP name ID* konfiguracji SAML ustaw:

```
urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified
```

MFA

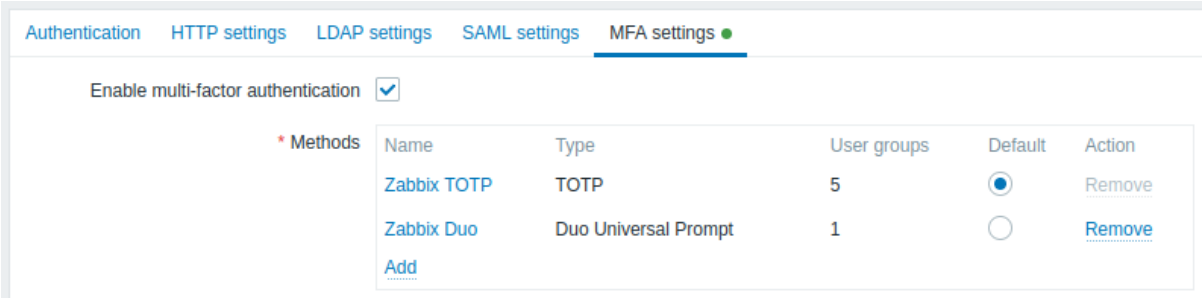
Przegląd

Uwierzytelnianie wieloskładnikowe authentication (MFA) może być używane do logowania do Zabbix, zapewniając dodatkową warstwę bezpieczeństwa poza samą nazwą użytkownika i hasłem.

W przypadku MFA użytkownik musi istnieć w Zabbix, musi podać poświadczenia Zabbix podczas logowania, a także musi potwierdzić swoją tożsamość w inny sposób, zazwyczaj za pomocą kodu wygenerowanego przez aplikację uwierzytelniającą na telefonie użytkownika.

Dostępnych jest wiele metod MFA, co pozwala użytkownikom wybrać opcję najlepiej odpowiadającą ich wymaganiom i preferencjom w zakresie bezpieczeństwa. Metody te to jednorazowe hasło czasowe (TOTP) oraz Duo Universal Prompt.

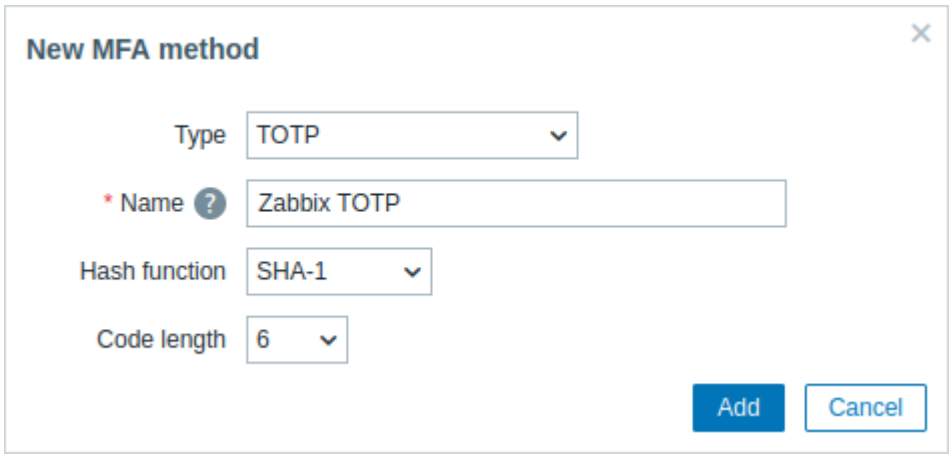
Konfiguracja



Parametry konfiguracji:

Parametr	Opis
Włącz uwierzytelnianie wieloskładnikowe	Zaznacz pole wyboru, aby włączyć uwierzytelnianie wieloskładnikowe.
Metody	Kliknij *Dodaj*, aby skonfigurować metodę MFA (zobacz konfigurację metody poniżej).

Konfiguracja metody



Parametry konfiguracji metody:

Parameter	Description
Type	Wybierz typ metody MFA: **TOTP** — użyj aplikacji uwierzytelniającej do generowania jednorazowych haseł opartych na czasie; **Duo Universal Prompt** — użyj usługi uwierzytelniania Duo do zapewnienia uwierzytelniania wieloskładnikowego.
Name	Wprowadź nazwę wyświetlaną jako nazwa konta dla wszystkich użytkowników MFA w aplikacjach uwierzytelniających (na przykład „Zabbix”).

Parameter	Description
Hash function	Wybierz funkcję skrótu (SHA-1, SHA-256 lub SHA-512) do generowania kodów TOTP. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „TOTP”. Należy pamiętać, że wybór SHA-256 lub SHA-512 może znacznie ograniczyć kompatybilność, ponieważ wiele aplikacji obecnie nie obsługuje tych funkcji.
Code length	Wybierz długość kodu weryfikacyjnego (6 lub 8). Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „TOTP”.
API hostname	Wprowadź nazwę hosta API podaną przez usługę uwierzytelniania Duo. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „Duo Universal Prompt”.
Client ID	Wprowadź identyfikator klienta podany przez usługę uwierzytelniania Duo. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „Duo Universal Prompt”.
Client secret	Wprowadź klucz tajny klienta podany przez usługę uwierzytelniania Duo. Ten parametr jest dostępny, jeśli typ metody MFA jest ustawiony na „Duo Universal Prompt”.

Przykłady konfiguracji

Ta sekcja zawiera przykłady konfiguracji MFA z użyciem Time-Based One-Time Password (TOTP) oraz Duo Universal Prompt.

TOTP

W przypadku TOTP użytkownicy muszą zweryfikować swoją tożsamość za pomocą aplikacji uwierzytelniającej (na przykład aplikacji Google Authenticator).

1. Przejdź do ustawień MFA w Zabbix w sekcji *Users* > *Authentication* i włącz uwierzytelnianie wieloskładnikowe.

2. Dodaj nową metodę MFA method z następującą konfiguracją:

   - Type: TOTP
   - Name: Zabbix TOTP
   - Hash function: SHA-1
   - Code length: 6

3. Kliknij przycisk *Add*, a następnie przycisk *Update*.

4. Przejdź do *Users* > *User groups* i utwórz nową grupę użytkowników z następującą configuration:

   - Group name: TOTP group
   - Users: Admin
   - Multi-factor authentication: Default (lub ”Zabbix TOTP”, jeśli nie jest ustawione jako domyślne)

5. Wyloguj się z Zabbix i zaloguj ponownie, używając swoich danych uwierzytelniających. Po pomyślnym zalogowaniu zostaniesz poproszony o rejestrację w MFA, z wyświetleniem kodu QR i klucza tajnego.



6. Zeskanuj kod QR lub wprowadź klucz tajny do aplikacji Google Authenticator. Aplikacja wygeneruje kod weryfikacyjny, który należy wpisać, aby zakończyć proces logowania.

7. Podczas kolejnych logowań pobierz kod weryfikacyjny z aplikacji Google Authenticator i wpisz go podczas logowania.

Duo Universal Prompt

W przypadku Duo Universal Prompt użytkownicy muszą potwierdzić swoją tożsamość za pomocą aplikacji uwierzytelniającej Duo Mobile.

Attention:
Metoda MFA Duo Universal Prompt wymaga zainstalowania rozszerzenia php-curl, dostępu do Zabbix przez HTTPS oraz □□□□□□ na połączenia wychodzące do serwerów Duo. Ponadto, jeśli włączono Content Security Policy (CSP) na serwerze WWW, należy dodać ”duo.com” do dyrektywy CSP w pliku konfiguracyjnym hosta wirtualnego.

1. Zarejestruj bezpłatne konto administratora Duo w Duo Signup.

2. Otwórz panel administracyjny Duo, przejdź do *Applications* > *Protect an Application*, wyszukaj aplikację *Web SDK* i kliknij *Protect*.

3. Zanotuj dane uwierzytelniające (Client ID, Client secret, API hostname) wymagane do skonfigurowania metody MFA w Zabbix.

4. Przejdź do ustawień MFA w Zabbix w sekcji *Users* > *Authentication* i włącz uwierzytelnianie wieloskładnikowe.

5. Dodaj nową metodę MFA z następującą konfiguracją:

   - Type: Duo Universal Prompt
   - Name: Zabbix Duo
   - API hostname: (użyj API hostname z Duo)
   - Client ID: (użyj Client ID z Duo)
   - Client secret: (użyj Client secret z Duo)

6. Kliknij przycisk *Add*, a następnie przycisk *Update*.

7. Przejdź do *Users* > *User groups* i utwórz nową grupę użytkowników z następującą konfiguracją:

   - Group name: Duo group
   - Users: Admin
   - Multi-factor authentication: Default (lub ”Zabbix Duo”, jeśli nie jest ustawione jako domyślne)

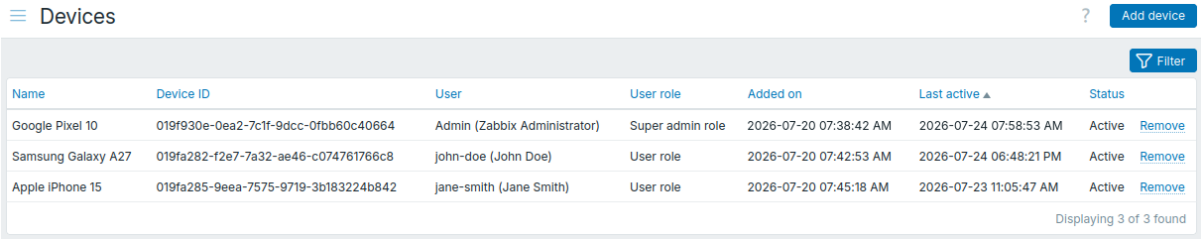
8. Wyloguj się z Zabbix i zaloguj ponownie przy użyciu swoich danych uwierzytelniających. Po pomyślnym zalogowaniu zostaniesz poproszony o rejestrację w MFA i przekierowany do Duo. Dokończ konfigurację Duo i zweryfikuj użytkownika za pomocą aplikacji Duo na telefonie, aby się zalogować.

9. Przy kolejnych logowaniach użyj odpowiedniej metody MFA udostępnionej przez aplikację Duo (na przykład pobrania kodu weryfikacyjnego, odpowiedzi na powiadomienia push lub użycia kluczy sprzętowych) i wprowadź wymagane informacje podczas logowania.

Urządzenia

Ta sekcja umożliwia administratorom najwyższego poziomu zarządzanie urządzeniami mobilnymi używanymi z aplikacją mobilną Zabbix.

Ta sekcja jest wyświetlana, jeśli włączono `EnableMobileDevices`.



Aby dodać urządzenie:

1. Kliknij *Dodaj urządzenie*.
2. Wybierz użytkownika, a następnie kliknij *Dodaj*. Użytkownik musi mieć włączony dostęp do urządzeń.
3. Zostanie wyświetlony kod QR. Użytkownik musi zeskanować ten kod w aplikacji mobilnej Zabbix.

Note:
Po dodaniu urządzenia dla użytkownika Zabbix dodaje również typ nośnika powiadomień push do nośników użytkownika. Po usunięciu urządzenia ten nośnik użytkownika również zostaje usunięty. Jeśli skonfigurowano wiele typów nośników powiadomień push, Zabbix dodaje ten, który jest włączony i znajduje się pierwszy w kolejności alfabetycznej.

Dodane urządzenia są wyświetlane na liście, którą można filtrować.

Można również usuwać urządzenia. Po usunięciu urządzenia jego dane uwierzytelniające dostępu zostają unieważnione i usunięte z Zabbix, po czym urządzenie nie może już uzyskać dostępu do Zabbix.

Lista zawiera następujące informacje:

Kolumna	Opis
Nazwa	Nazwa urządzenia.
Identyfikator urządzenia	Identyfikator urządzenia.
Użytkownik	Użytkownik, dla którego dodano urządzenie, lub *Niedostępny użytkownik*, jeśli ten użytkownik został usunięty.
Rola użytkownika	Rola użytkownika.
Dodano	Data i godzina dodania urządzenia.
Ostatnia aktywność	Data i godzina ostatniej aktywności urządzenia.
Stan	Stan urządzenia: **Nowe** - urządzenie zostało dodane bez zeskanowania kodu QR i będzie trzeba dodać je ponownie. **Nowe (token wygasł)** - urządzenie zostało dodane bez zeskanowania kodu QR, a token w kodzie QR wygasł; urządzenie będzie trzeba dodać ponownie. **Aktywne** - urządzenie zostało dodane, kod QR został zeskanowany i może korzystać z aplikacji mobilnej Zabbix. **Nieprzypisane** - urządzenie zostało dodane, ale jego użytkownik został usunięty i urządzenie nie może korzystać z aplikacji mobilnej Zabbix.

Użytkownicy z wystarczającymi uprawnieniami mogą również zarządzać własnymi urządzeniami w sekcji *Ustawienia użytkownika* > *Urządzenia*.

Administracja

Przegląd

Menu Administracja służy do funkcji administracyjnych Zabbix. To menu jest dostępne wyłącznie dla użytkowników typu SuperAdmin.

Źródło danych

Przegląd

Sekcja *Administracja* > *Źródło danych* umożliwia określenie źródła danych telemetrycznych, czyli bazy danych przechowującej dane telemetryczne (ślady, metryki i logi). Dane telemetryczne są używane do monitorowania wydajności aplikacji (APM), co pomaga sprawdzać przepływ żądań przez aplikacje i usługi, mierzyć ich wydajność oraz znajdować przyczynę powolnych lub nieudanych operacji.

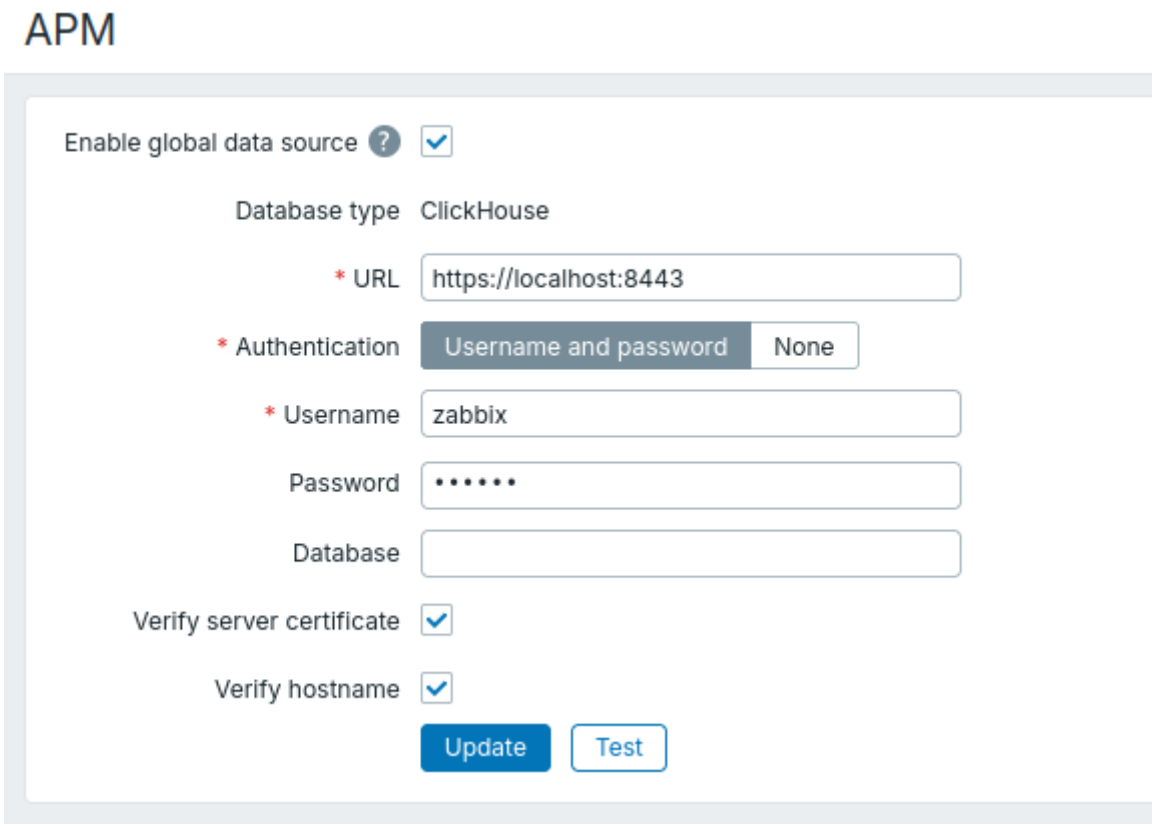
Źródło danych telemetrycznych musi obsługiwać OpenTelemetry.

Źródło danych telemetrycznych określone w tej sekcji jest nazywane globalnym źródłem danych, ponieważ jest współdzielone przez serwer Zabbix, proxy i frontend. Można jednak zastąpić je dla dowolnego z tych komponentów, określając inne źródło danych telemetrycznych w konfiguracji serwera, proxy lub frontendu.

Pozycja zapytanie telemetryczne odpytuje to źródło danych, chyba że w konfiguracji komponentu określono inne źródło.

APM

Aby określić globalne źródło danych, zaznacz pole wyboru *Włącz globalne źródło danych* i wypełnij opisane poniżej parametry. Przed zapisaniem możesz kliknąć *Testuj*, aby sprawdzić połączenie z tym źródłem danych (wymaga rozszerzenia php-curl).



Parametr	Opis
Typ bazy danych	Typ bazy danych jest na stałe ustawiony na *ClickHouse* i nie można go zmienić. Zobacz także: Wymagania oraz Konfiguracja ClickHouse.
URL	Wprowadź adres URL HTTP lub HTTPS używany do łączenia się z bazą danych.
Uwierzytelnianie	Wybierz metodę uwierzytelniania używaną do logowania się do bazy danych: *Nazwa użytkownika i hasło* lub *Brak*.
Nazwa użytkownika	Określ nazwę użytkownika używaną do logowania się do bazy danych. Ten parametr jest dostępny, jeśli dla opcji *Uwierzytelnianie* ustawiono wartość *Nazwa użytkownika i hasło*.
Hasło	Określ hasło używane do logowania się do bazy danych lub pozostaw pole puste, jeśli użytkownik bazy danych nie ma hasła. Hasło jest resetowane po zmianie adresu URL. Ten parametr jest dostępny, jeśli dla opcji *Uwierzytelnianie* ustawiono wartość *Nazwa użytkownika i hasło*.
Baza danych	Wprowadź nazwę bazy danych zawierającej dane telemetryczne. Jeśli pole pozostanie puste, Zabbix użyje domyślnej bazy danych skonfigurowanej w ClickHouse dla użytkownika.
Weryfikuj certyfikat serwera	Zaznacz pole wyboru, aby zweryfikować certyfikat serwera. Jeśli pole jest zaznaczone, powiązane opcje TLS muszą być ustawione w konfiguracji serwera, proxy lub frontend. Ten parametr jest dostępny, jeśli opcja *URL* zawiera adres URL HTTPS.
Weryfikuj nazwę hosta	Zaznacz pole wyboru, aby sprawdzić, czy nazwa hosta w opcji *URL* odpowiada nazwie hosta w certyfikacie serwera. Jeśli pole jest zaznaczone, powiązane opcje TLS muszą być ustawione w konfiguracji serwera, proxy lub frontend. Ten parametr jest dostępny, jeśli opcja *URL* zawiera adres URL HTTPS.

Ogólne

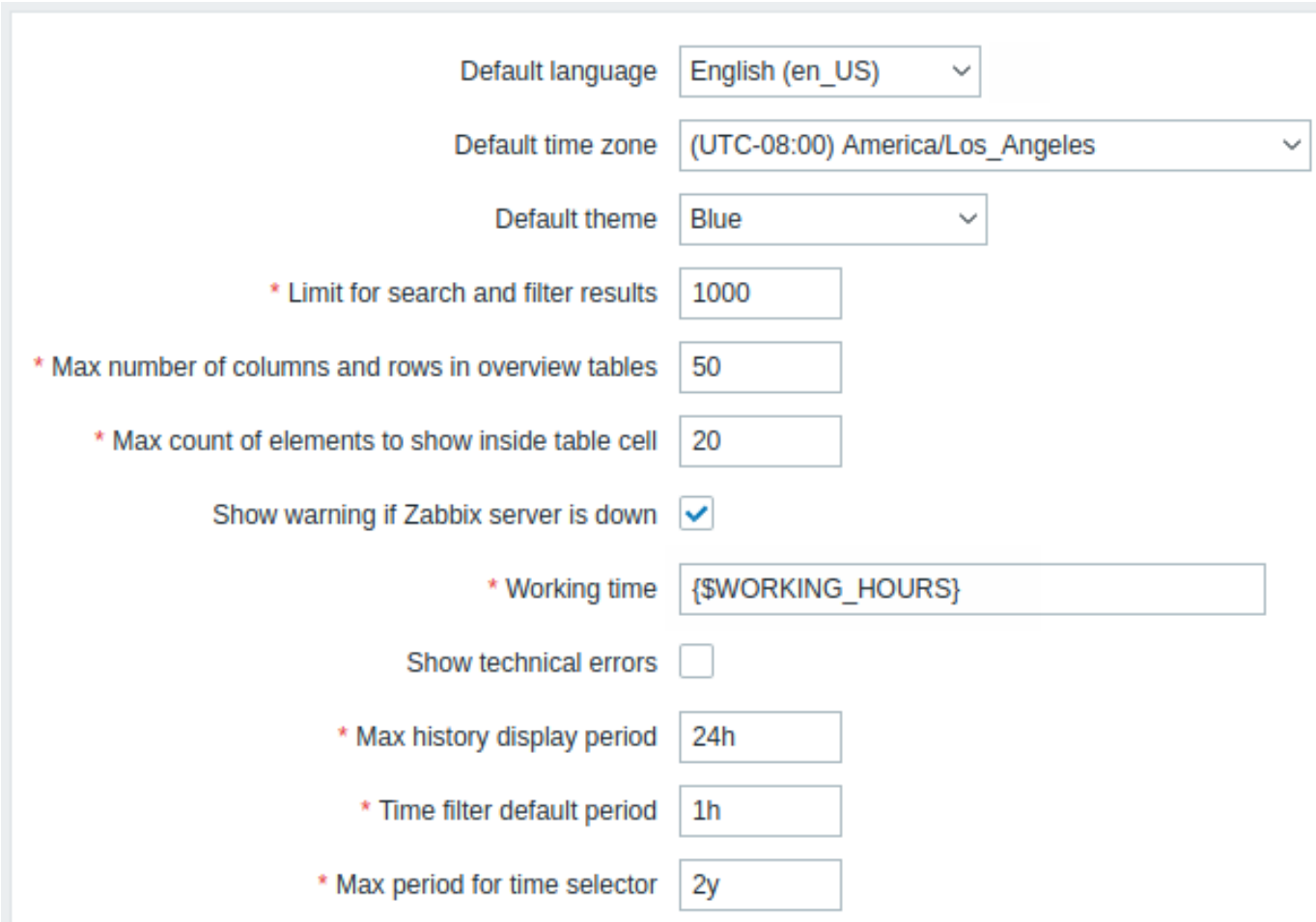
Przegląd

Sekcja *Administration* > *General* zawiera kilka podsekcji służących do ustawiania domyślnych wartości związanych z frontendem oraz dostosowywania Zabbixa.

Lista dostępnych podsekcji pojawia się po kliknięciu *General* w sekcji menu *Administration*. Możliwe jest również przełączanie się między podsekcjami za pomocą listy rozwijanej tytułu w lewym górnym rogu.

GUI

Ta sekcja umożliwia dostosowanie kilku domyślnych ustawień związanych z frontendem.



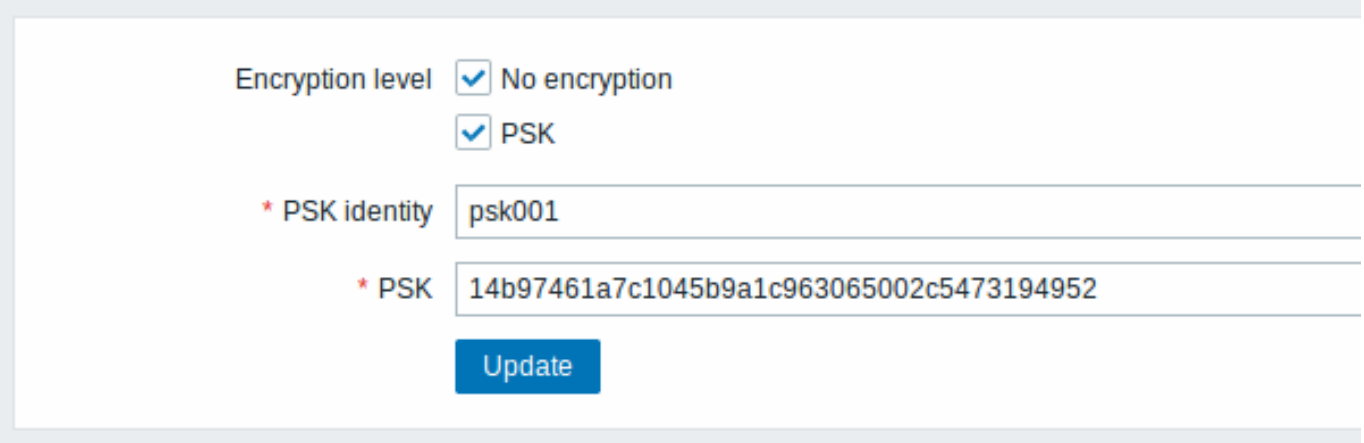
Parametry konfiguracji:

Parameter	Description
Default language	Domyślny język dla użytkowników, którzy nie określili języka w swoich profilach, oraz dla użytkowników gościnnych. Więcej informacji można znaleźć w sekcji Instalacja dodatkowych języków frontend.
Default time zone	Domyślna strefa czasowa dla użytkowników, którzy nie określili strefy czasowej w swoich profilach, oraz dla użytkowników gościnnych.
Default theme	Domyślny motyw dla użytkowników, którzy nie określili motywu w swoich profilach, oraz dla użytkowników gościnnych.
Limit for search and filter results	Maksymalna liczba elementów (wierszy), które będą wyświetlane na liście w interfejsie webowym, na przykład w *Zbieranie danych* > *Hosty*. *Uwaga*: jeśli ustawiono na przykład wartość „50”, we wszystkich objętych tym ustawieniem listach frontend będą wyświetlane tylko pierwsze 50 elementów. Jeśli jakaś lista zawiera więcej niż pięćdziesiąt elementów, zostanie to oznaczone znakiem „+” w komunikacie *”Wyświetlanie od 1 do 50 z **50+** znalezionych”*. Ponadto, jeśli używane jest filtrowanie i nadal istnieje więcej niż 50 dopasowań, wyświetlonych zostanie tylko pierwszych 50. Należy pamiętać, że zwiększenie wartości tego parametru może prowadzić do pogorszenia wydajności i zwiększonego zużycia pamięci po stronie serwera WWW.
Max number of columns and rows in overview tables	Maksymalna liczba kolumn i wierszy wyświetlanych w widżecie pulpitu *Trigger overview*. Ten sam limit dotyczy zarówno kolumn, jak i wierszy. Jeśli istnieje więcej wierszy i/lub kolumn niż wyświetlanych, system pokaże ostrzeżenie na dole tabeli: ”Nie są wyświetlane wszystkie wyniki. Podaj bardziej szczegółowe kryteria wyszukiwania.”
Max count of elements to show inside table cell	W przypadku wpisów wyświetlanych w pojedynczej komórce tabeli zostanie pokazana nie większa liczba elementów niż skonfigurowana tutaj.

Parameter	Description
Show warning if Zabbix server is down	Ten parametr włącza wyświetlanie komunikatu ostrzegawczego w oknie przeglądarki, jeśli nie można połączyć się z serwerem Zabbix (możliwa awaria). Komunikat pozostaje widoczny nawet podczas przewijania strony. Po najechaniu kursorem jest tymczasowo ukrywany, aby odsłonić treść znajdującą się pod nim.
Working time	Ten parametr obowiązujący w całym systemie definiuje godziny pracy. Na wykresach czas pracy jest wyświetlany jako białe tło, a czas poza pracą jako szary. Opis formatu czasu znajduje się na stronie Specyfikacja okresu czasu. Obsługiwane są makra użytkownika.
Show technical errors	Jeśli zaznaczone, wszyscy zarejestrowani użytkownicy będą mogli widzieć błędy techniczne (PHP/SQL). Jeśli odznaczone, informacje te są dostępne tylko dla Zabbix Super Admins oraz użytkowników należących do grup użytkowników z włączonym trybem debugowania.
Max history display period	Maksymalny okres, dla którego będą wyświetlane dane historyczne w *Monitoring* > *Latest data*, informacje o scenariuszu web hosta w *Monitoring* > *Hosts* oraz w widżecie pulpitu *Top items*. Dozwolony zakres: 24 godziny (domyślnie) - 1 tydzień. Obsługiwane są sufiksy czasu, np. 1w (jeden tydzień), 36h (36 godzin).
Time filter default period	Domyślny okres czasu używany na wykresach i pulpitach. Dozwolony zakres: 1 minuta - 10 lat (domyślnie: 1 godzina). Obsługiwane są sufiksy czasu, np. 10m (dziesięć minut), 5w (pięć tygodni). Uwaga: gdy użytkownik zmieni okres czasu podczas przeglądania wykresu, okres ten jest zapisywany jako preferencja użytkownika, zastępując globalną wartość domyślną lub wcześniejszy wybór użytkownika.
Max period for time selector	Maksymalny dostępny okres czasu dla wykresów i pulpitów. Użytkownicy nie będą mogli wizualizować starszych danych. Dozwolony zakres: 1 rok - 10 lat (domyślnie: 2 lata). Obsługiwane są sufiksy czasu, np. 1y (jeden rok), 365w (365 tygodni).

Autorejestracja

W tej sekcji można skonfigurować poziom szyfrowania dla autorejestracji aktywnego agent.



Parametry oznaczone gwiazdką są obowiązkowe.

Parametry konfiguracji:

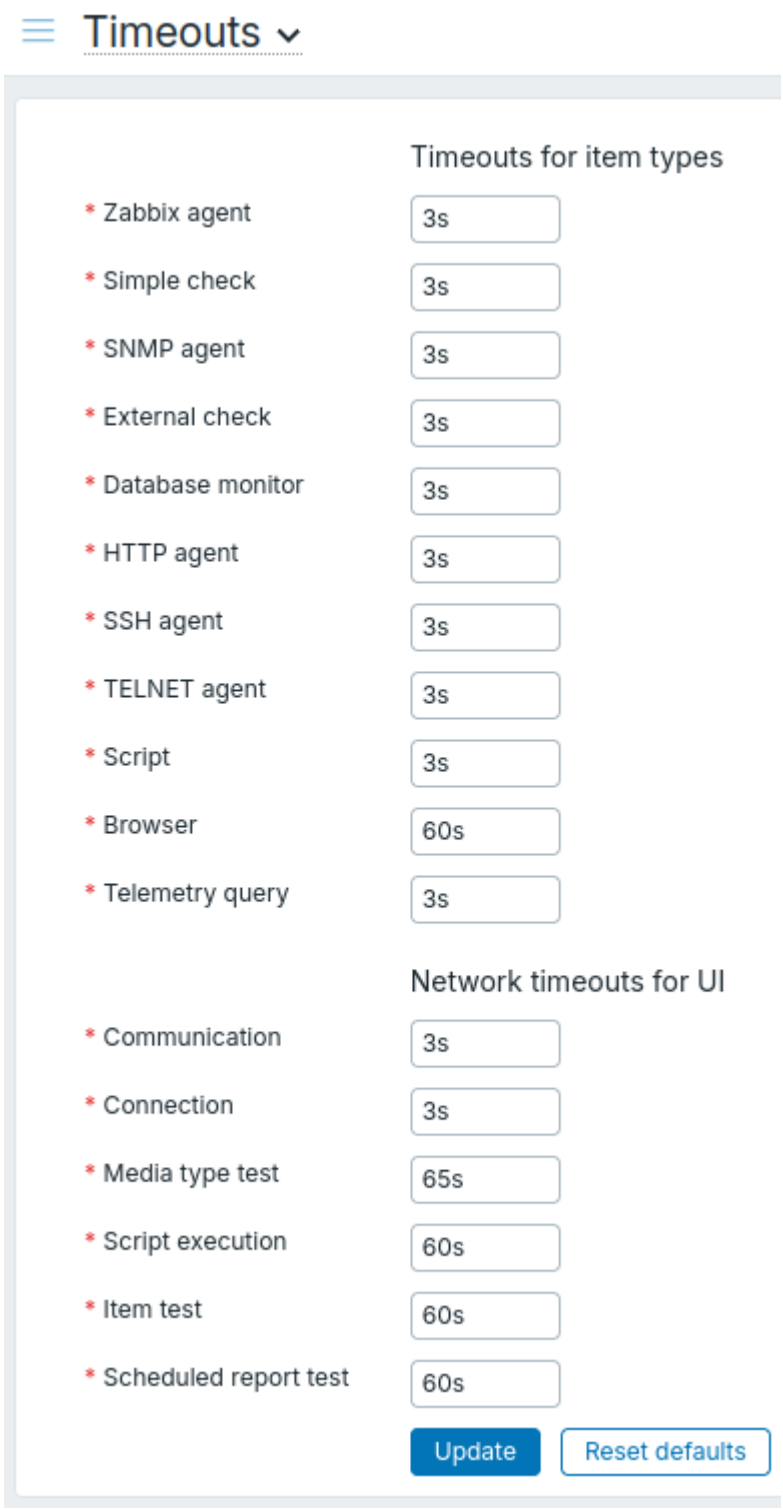
Parameter	Description
Poziom szyfrowania	Wybierz jedną lub obie opcje poziomu szyfrowania: **Brak szyfrowania** - dozwolone są nieszyfrowane połączenia **PSK** - dozwolone są szyfrowane połączenia TLS z użyciem klucza wstępnie współdzielonego
Tożsamość PSK	Wprowadź ciąg tożsamości klucza wstępnie współdzielonego. To pole jest dostępne tylko wtedy, gdy jako *Poziom szyfrowania* wybrano opcję 'PSK'. Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK użyć.

Parameter	Description
PSK	Wprowadź klucz wstępnie współdzielony (parzysta liczba znaków szesnastkowych). Maksymalna długość: 512 cyfr szesnastkowych (PSK o rozmiarze 256 bajtów), jeśli Zabbix używa biblioteki GnuTLS lub OpenSSL, 64 cyfry szesnastkowe (PSK o rozmiarze 32 bajtów), jeśli Zabbix używa biblioteki mbed TLS (PolarSSL). Przykład: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952 To pole jest dostępne tylko wtedy, gdy jako *Poziom szyfrowania* wybrano opcję 'PSK'.

Zobacz także: Secure autoregistration

Limity czasu

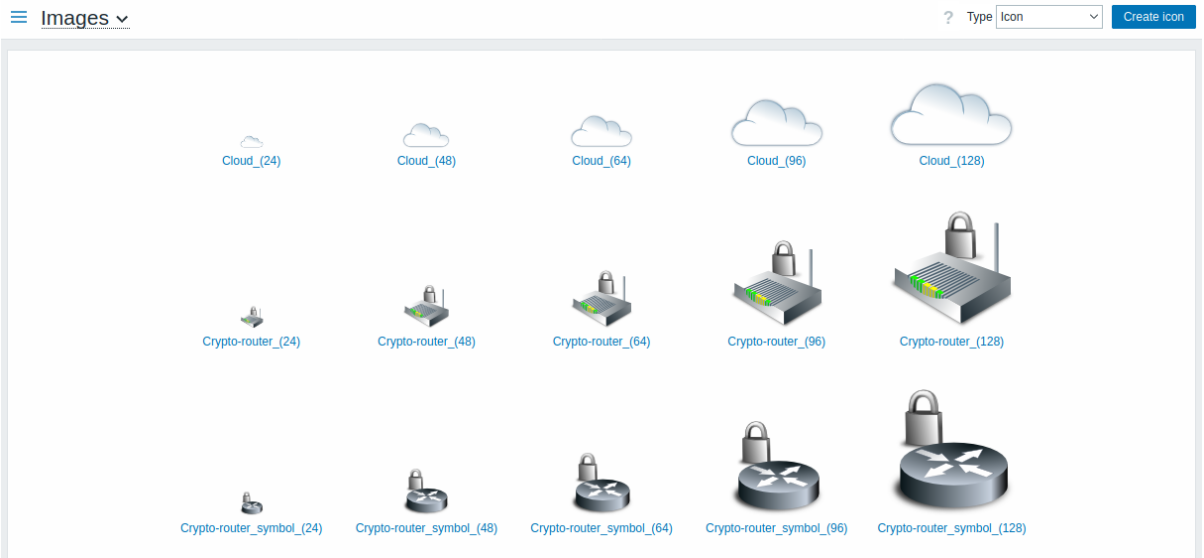
W tej sekcji można ustawić globalne limity czasu dla typów pozycji oraz limity czasu sieci. Wszystkie pola w tym formularzu są wymagane.



Parameter		Description
Limity czasu dla typów pozycji		Ile sekund Zabbix ma czekać przed przerwaniem sprawdzenia (w zależności od jego typu). Dodatkowe informacje i ograniczenia można znaleźć w sekcji item timeout. Dozwolony zakres: 1 - 600 s (domyślnie: 3 s; domyślnie dla pozycji browser: 60 s). Obsługiwane są sufiksy czasu, np. 30s, 1m, oraz makra użytkownika. Należy pamiętać, że limity czasu na poziomie proxy (timeouts), jeśli są skonfigurowane, zastępują te ustawienia globalne. Ponadto, jeśli skonfigurowano limity czasu dla określonych pozycji, zastępują one zarówno ustawienia globalne, jak i ustawienia proxy.
Limity czasu sieci dla UI		
	Komunikacja	Ile sekund czekać przed zamknięciem bezczynnego gniazda (jeśli połączenie z serwerem Zabbix zostało wcześniej nawiązane, ale frontend nie może zakończyć operacji odczytu/wysyłania danych w tym czasie, połączenie zostanie przerwane). Dozwolony zakres: 1 - 300 s (domyślnie: 3 s).
	Połączenie	Ile sekund czekać przed przerwaniem próby połączenia z serwerem Zabbix. Dozwolony zakres: 1 - 30 s (domyślnie: 3 s).
	Test typu media	Ile sekund czekać na odpowiedź podczas testowania typu media. Dozwolony zakres: 1 - 300 s (domyślnie: 65 s).
	Wykonanie skryptu	Ile sekund czekać na odpowiedź podczas wykonywania skryptu. Dozwolony zakres: 1 - 300 s (domyślnie: 60 s). Ten limit czasu dotyczy całego łańcucha wykonywania skryptu, który może mieć różną długość. Na przykład, jeśli skrypt jest wykonywany na agencie, jest to przejście przez serwer (ewentualnie także proxy) do agenta i z powrotem. W przypadku skryptów wykonywanych na agencie obowiązuje limit czasu serwer lub proxy. W przypadku skryptów wykonywanych wyłącznie na aktywnym agencie prawdopodobne jest, że domyślny limit czasu serwera/proxy musi zostać zwiększony. Limit czasu serwera/proxy musi być większy niż częstotliwość odświeżania aktywnego sprawdzenia, w przeciwnym razie może zostać przekroczony, zanim aktywny agent w ogóle otrzyma skrypt. Zobacz też: Script timeout.
	Test pozycji	Ile sekund czekać na zwrócone dane podczas testowania pozycji. Dozwolony zakres: 1 - 600 s (domyślnie: 60 s).
	Test raportu zaplanowanego	Ile sekund czekać na zwrócone dane podczas testowania zaplanowanego raportu. Dozwolony zakres: 1 - 300 s (domyślnie: 60 s).

Obrazy

Sekcja Obrazy wyświetla wszystkie obrazy dostępne w Zabbix. Obrazy są przechowywane w bazie danych.

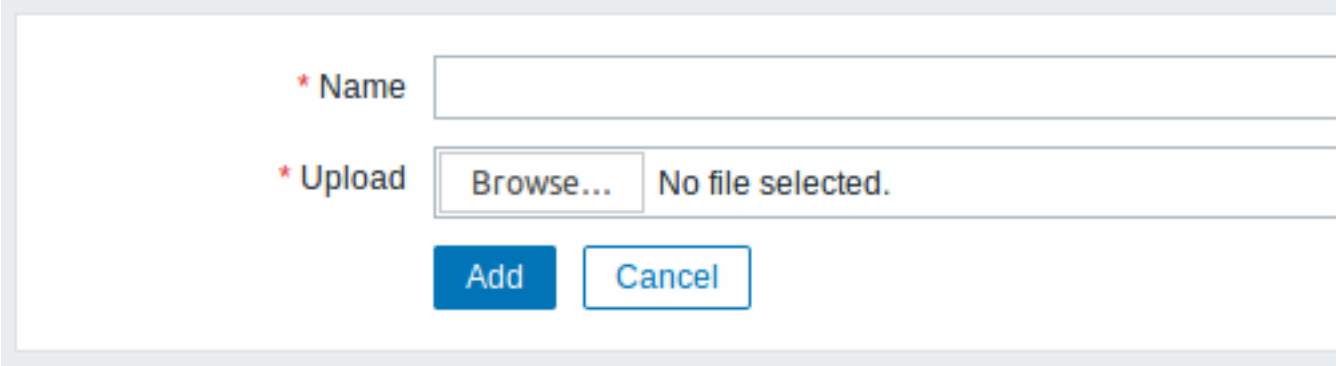


Lista rozwijana *Typ* umożliwia przełączanie między obrazami ikon i tła:

- Ikony służą do wyświetlania elementów mapy sieciowej.
- Obrazy tła są używane jako tło map sieciowych.

Dodawanie obrazu

Możesz dodać własny obraz, klikając przycisk *Utwórz ikonę* lub *Utwórz tło* w prawym górnym rogu.



Atrybuty obrazu:

Parametr	Opis
Nazwa	Unikalna nazwa obrazu.
Prześlij	Wybierz plik (PNG, JPEG, GIF lub WebP) z systemu lokalnego, który ma zostać przesłany do Zabbix. *Uwaga*: możliwe jest przesłanie innych formatów, które podczas przesyłania zostaną przekonwertowane do formatu PNG. Do przetwarzania obrazów używana jest biblioteka GD, dlatego obsługiwane formaty zależą od używanej wersji biblioteki (Zabbix wymaga wersji 2.0.28 lub nowszej).

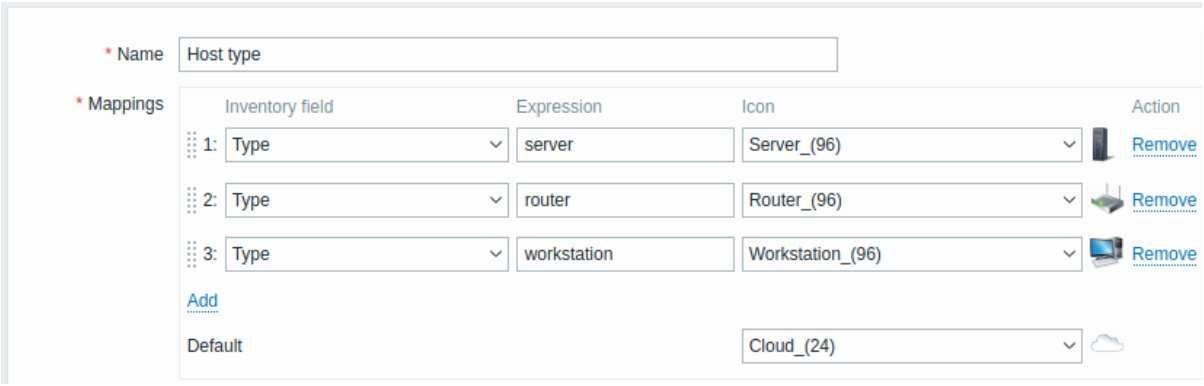
Note:
Maksymalny rozmiar przesyłanego pliku jest ograniczony wartością `ZBX_MAX_IMAGE_SIZE`, która wynosi 1024x1024 bajtów lub 1 MB.
Przesłanie obrazu może się nie powieść, jeśli jego rozmiar jest zbliżony do 1 MB, a parametr konfiguracyjny MySQL `max_allowed_packet` ma domyślną wartość 1 MB. W takim przypadku zwiększ wartość parametru `max_allowed_packet`.

Mapowanie ikon

Ta sekcja umożliwia tworzenie mapowania określonych hostów z określonymi ikonami. Do utworzenia mapowania wykorzystywane są informacje z pól inwentarza hosta.

Mapowania można następnie wykorzystać w konfiguracji mapy sieciowej, aby automatycznie przypisywać odpowiednie ikony do pasujących hostów.

Aby utworzyć nową mapę ikon, kliknij *Create icon map* w prawym górnym rogu.



Parametry konfiguracji:

Parameter	Description
Name	Unikalna nazwa mapy ikon.
Mappings	Lista mapowań. Kolejność mapowań określa, które z nich będzie miało priorytet. Możesz przenosić mapowania w górę i w dół listy metodą przeciągnij i upuść.

Parameter	Description
Inventory field	Pole inwentarza hosta, które będzie sprawdzane w celu znalezienia dopasowania.
Expression	Wyrażenie regularne opisujące dopasowanie.
Icon	Ikona używana, jeśli zostanie znalezione dopasowanie do wyrażenia.
Default	Domyślna ikona do użycia.

Wyrażenia regularne

Ta sekcja umożliwia tworzenie niestandardowych wyrażeń regularnych, które można używać w kilku miejscach w frontend. Szczegółowe informacje można znaleźć w sekcji Wyrażenia regularne.

Opcje wyświetlania wyzwalaczy

Ta sekcja umożliwia dostosowanie sposobu wyświetlania stanu wyzwalacza we frontendzie oraz nazw i kolorów poziomu wyzwalacza.

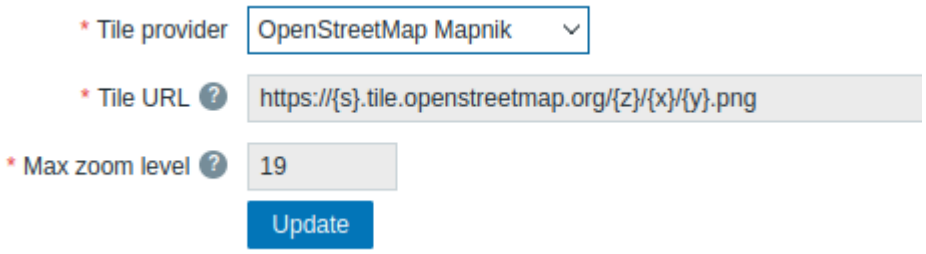


Parameter	Description
Use custom event status colors	Zaznaczenie tego parametru włącza dostosowywanie kolorów dla problemów potwierdzonych i niepotwierdzonych.
Unacknowledged PROBLEM events, *Acknowledged PROBLEM events*, *Unacknowledged RESOLVED events*, *Acknowledged RESOLVED events*	Wprowadź nowy kod koloru lub kliknij kolor, aby wybrać nowy z dostępnej palety. Jeśli zaznaczono pole wyboru *blinking*, wyzwalacze będą migać przez pewien czas po zmianie stanu, aby były bardziej widoczne.
Display OK triggers for	Okres wyświetlania wyzwalaczy OK. Dozwolony zakres: 0 - 24 godziny. Obsługiwane są sufiksy czasu, np. 5m, 2h, 1d.
On status change triggers blink for	Czas migania wyzwalacza. Dozwolony zakres: 0 - 24 godziny. Obsługiwane są sufiksy czasu, np. 5m, 2h, 1d.

Parameter	Description
Not classified, *Information*, *Warning*, *Average*, *High*, *Disaster*	Niestandardowe nazwy poziomów ważności i/lub kolory wyświetlane zamiast domyślnych systemowych. Wprowadź nowy kod koloru lub kliknij kolor, aby wybrać nowy z dostępnej palety. Pamiętaj, że niestandardowe nazwy poziomów ważności wprowadzone tutaj będą używane we wszystkich lokalizacjach. Jeśli musisz przetłumaczyć je na inne języki dla określonych użytkowników, zobacz stronę Dostosowywanie poziomów ważności wyzwalaczy.

Mapy geograficzne

Ta sekcja umożliwia wybór dostawcy usługi kafelków map geograficznych oraz konfigurację ustawień dostawcy usług dla widżetu dashboardu Geomap. Aby zapewnić wizualizację z użyciem map geograficznych, Zabbix korzysta z otwartoźródłowej biblioteki JavaScript do interaktywnych map Leaflet. Należy pamiętać, że Zabbix nie ma wpływu na jakość obrazów dostarczanych przez zewnętrznych dostawców kafelków, w tym przez predefiniowanych dostawców kafelków.



Parameter	Description
Tile provider	Wybierz jednego z dostępnych dostawców usługi kafelków albo wybierz *Other*, aby dodać innego dostawcę kafelków lub kafelki hostowane lokalnie (zobacz Korzystanie z niestandardowego dostawcy usługi kafelków).
Tile URL	Szablon adresu URL (do 2048 znaków) służący do ładowania i wyświetlania warstwy kafelków na mapach geograficznych. To pole jest edytowalne tylko wtedy, gdy *Tile provider* ma wartość *Other*. Następujące symbole zastępcze są obsługiwane: *{s}* reprezentuje jedną z dostępnych subdomen; *{z}* reprezentuje parametr poziomu powiększenia w adresie URL; *{x}* i *{y}* reprezentują współrzędne kafelka; *{r}* może być użyte do dodania "@2x" do adresu URL w celu wczytania kafelków retina. Przykład: `https://{s}.example.com/{z}/{x}/{y}{r}.png`
Attribution text	Tekst atrybucji dostawcy kafelków, który ma być wyświetlany w małym polu tekstowym na mapie. To pole jest widoczne tylko wtedy, gdy *Tile provider* ma wartość *Other*.
Max zoom level	Maksymalny poziom powiększenia mapy. To pole jest edytowalne tylko wtedy, gdy *Tile provider* ma wartość *Other*.

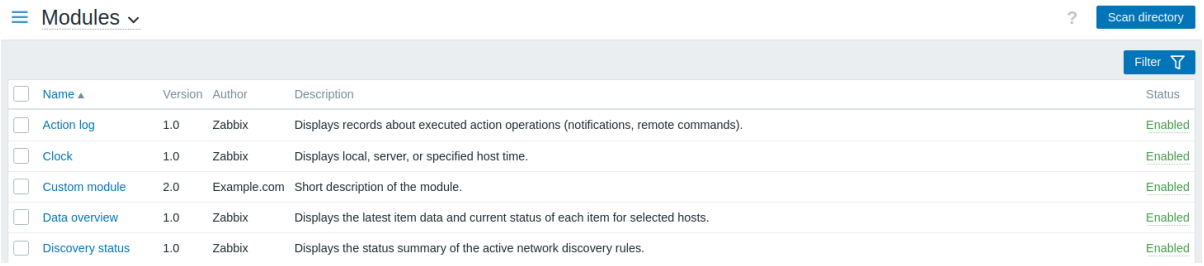
Korzystanie z niestandardowego dostawcy usługi kafelków

Widżet Geomap może wczytywać rastrowe obrazy kafelków z niestandardowej, hostowanej samodzielnie lub zewnętrznej usługi dostawcy kafelków. Aby użyć niestandardowej zewnętrznej usługi dostawcy kafelków albo samodzielnie hostowanego folderu lub serwera kafelków, wybierz *Other* w polu *Tile provider* i określ niestandardowy adres URL w polu *Tile URL*, używając odpowiednich symboli zastępczych.

Moduły

Ta sekcja umożliwia zarządzanie niestandardowymi oraz wbudowanymi modułami frontend.

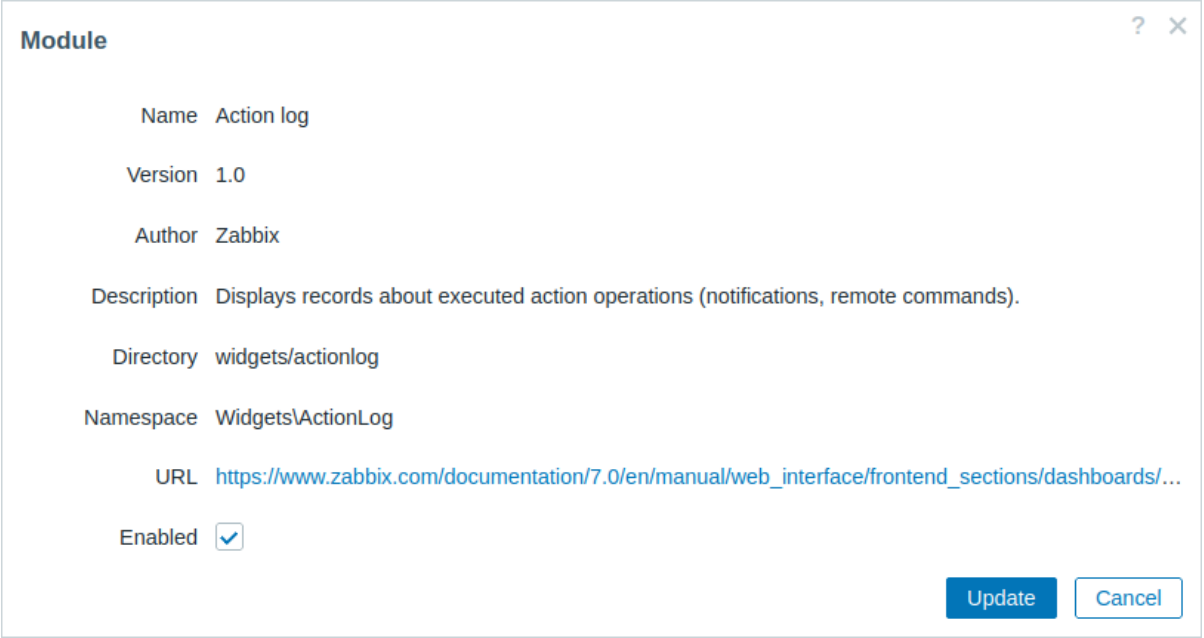
Note:
Jeśli flaga funkcji `$ZBX_FEATURE_FLAGS['modules_config_enabled']` ma wartość `false` w pliku konfiguracji frontend, dostęp do tej sekcji jest wyłączony.



Kliknij *Skanuj katalog*, aby zarejestrować lub wyrejestrować niestandardowe moduły. Zarejestrowane moduły pojawią się na liście, a wyrejestrowane moduły zostaną z niej usunięte.

Kliknij status modułu na liście, aby go włączyć lub wyłączyć. Możesz również masowo włączać lub wyłączać moduły, zaznaczając je na liście, a następnie klikając przyciski *Włącz/Wyłącz* poniżej listy.

Kliknij nazwę modułu na liście, aby wyświetlić jego szczegóły w wyskakującym oknie.

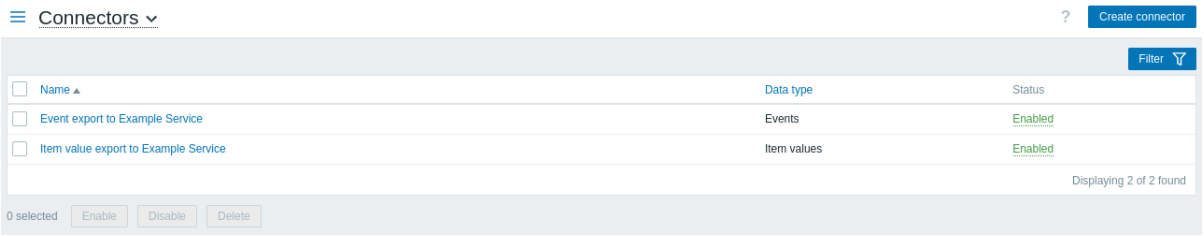


Status modułu można również zaktualizować w wyskakującym oknie ze szczegółami modułu. W tym celu zaznacz lub usuń zaznaczenie pola wyboru *Włączony*, a następnie kliknij *Aktualizuj*.

Możesz filtrować moduły według nazwy lub statusu (włączone/wyłączone).

Złącza

Ta sekcja umożliwia skonfigurowanie złączy do przesyłania danych Zabbix do systemów zewnętrznych za pośrednictwem protokołu HTTP.



Kliknij *Utwórz złącze*, aby skonfigurować nowe złącze.

Złącza można filtrować według nazwy lub statusu (włączone/wyłączone). Kliknij status złącza na liście, aby je włączyć/wyłączyć. Możesz także zbiorczo włączyć/wyłączyć złącza, zaznaczając je na liście, a następnie klikając przyciski *Włącz/Wyłącz* poniżej listy.

Inne

Ta sekcja umożliwia konfigurację różnych pozostałych parametrów frontend.



Parametr	Opis
URL frontend	Adres URL (do 2048 znaków) do interfejsu webowego Zabbix. Ten parametr jest używany przez usługę webową Zabbix do komunikacji z frontend i powinien zostać określony, aby włączyć raporty cykliczne.
Grupa dla wykrytych hostów	Hosty wykryte przez wykrywanie sieci i autorejestrację agent zostaną automatycznie umieszczone w wybranej tutaj grupie hostów.
Domyślny tryb inwentaryzacji hosta	Domyślny tryb inwentaryzacji hosta. Będzie stosowany zawsze, gdy nowy host lub prototyp hosta zostanie utworzony przez serwer lub frontend, chyba że zostanie nadpisany podczas wykrywania/autorejestracji hosta przez operację *Ustaw tryb inwentaryzacji hosta*.

Parametr	Opis
Grupa użytkowników dla komunikatu o niedostępności bazy danych	Grupa użytkowników do wysyłania komunikatu alarmowego o niedostępności bazy danych. Serwer Zabbix zależy od dostępności bazy danych backendu. Nie może działać bez bazy danych. Jeśli baza danych jest niedostępna, wybrani użytkownicy mogą zostać powiadomieni przez Zabbix. Powiadomienia będą wysyłane do ustawionej tutaj grupy użytkowników przy użyciu włączonych wpisów mediów użytkownika. Powiadomienia są przesyłane przy użyciu następujących typów mediów (jeśli są włączone): e-mail; SMS; niestandardowe skrypty alertów. Nawet jeśli wpis mediów typu webhook jest skonfigurowany i włączony, nie będzie wyzwalał powiadomień. Serwer Zabbix nie zatrzyma się; będzie czekał, aż baza danych ponownie stanie się dostępna, aby kontynuować przetwarzanie. Powiadomienie składa się z następującej treści: `[MySQL|PostgreSQL] database <DB Name> [on <DB Host>:<DB Port>] is not available: <error message depending on the type of DBMS (database)>` <DB Host> nie jest dodawany do komunikatu, jeśli jest zdefiniowany jako pusta wartość, a <DB Port> nie jest dodawany, jeśli ma wartość domyślną („0”). Menedżer alertów (specjalny proces serwera Zabbix) próbuje ustanowić nowe połączenie z bazą danych co 10 sekund. Jeśli baza danych nadal jest niedostępna, menedżer alertów ponawia wysyłanie alertów, ale nie częściej niż co 15 minut.
Rejestruj niedopasowane pułapki SNMP	Rejestruj pułapkę SNMP, jeśli nie znaleziono odpowiadających interfejsów SNMP.

Autoryzacja

Parametr	Opis
Próby logowania	Liczba nieudanych prób logowania, po której możliwość zalogowania się zostanie zablokowana.
Interwał blokady logowania	Okres, przez który logowanie będzie zabronione po przekroczeniu limitu *Próby logowania*. Dozwolony zakres: 30 - 3600 sekund. Obsługiwane są sufiksy czasu, np. 90s, 5m, 1h.

Przechowywanie sekretów

Parameter	Description
Vault provider	Wybierz oprogramowanie do zarządzania sekretami do przechowywania wartości user macro - *HashiCorp Vault* (domyślnie) lub *CyberArk Vault*.
Resolve secret vault macros by	Rozwiązywanie makr sejfu sekretów przez: **serwer Zabbix** - sekrety są pobierane przez serwer Zabbix i w razie potrzeby przekazywane do proxy (domyślnie); **serwer Zabbix i proxy** - sekrety są pobierane zarówno przez serwer Zabbix, jak i proxy, co umożliwia im niezależne rozwiązywanie makr.

Zobacz też: Przechowywanie sekretów.

Bezpieczeństwo

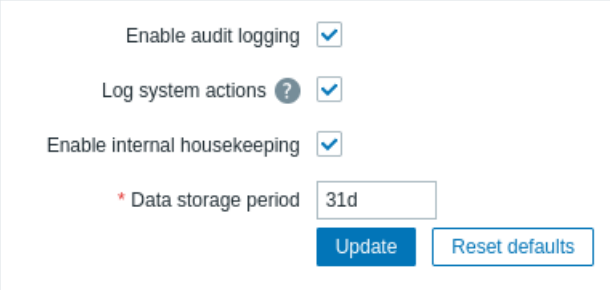
Parametr	Opis
Validate URI schemes	Odznacz to pole wyboru, aby wyłączyć walidację schematów URI (domyślnie włączona). Jeśli jest zaznaczone, możesz określić rozdzieloną przecinkami listę dozwolonych schematów URI (domyślnie: http,https,ftp,file,mailto,tel,ssh). Dotyczy wszystkich pól we frontend, w których używane są URI (na przykład adresów URL elementów mapy).

Parametr	Opis
Use X-Frame-Options HTTP header	Odznacz to pole wyboru, aby wyłączyć nagłówek HTTP X-Frame-options (niezalecane). Jeśli jest zaznaczone, możesz określić wartość nagłówka HTTP X-Frame-options. Obsługiwane wartości: **SAMEORIGIN** (domyślnie) lub **'self'** (musi być ujęte w pojedyncze cudzysłowy) - strona może być wyświetlana w ramce tylko w tym samym źródle co sama strona; **DENY** lub **'none'** (musi być ujęte w pojedyncze cudzysłowy) - strona nie może być wyświetlana w ramce, niezależnie od witryny próbującej to zrobić; **ciąg nazw hostów oddzielonych spacjami**; dodanie **'self'** (musi być ujęte w pojedyncze cudzysłowy) do listy pozwala na wyświetlanie strony w ramce w tym samym źródle co sama strona. Zwróć uwagę, że użycie **'self'** lub **'none'** bez pojedynczych cudzysłowów spowoduje potraktowanie ich jako nazw hostów.
Use iframe sandboxing	Odznacz to pole wyboru, aby wyłączyć umieszczanie zawartości pobranego adresu URL w sandboxie (niezalecane). Jeśli jest zaznaczone, możesz określić wyjątki sandboxingu iframe; nieokreślone ograniczenia nadal będą stosowane. Jeśli to pole jest puste, zastosowanie mają wszystkie ograniczenia atrybutu sandbox. Aby uzyskać więcej informacji, zobacz opis atrybutu `sandbox`.

Dziennik audytu

Przegląd

Ta sekcja umożliwia konfigurację ustawień dziennika audytu.



Dostępne są następujące parametry:

Parametr	Opis
Włącz rejestrowanie audytu	Włącz (domyślnie) lub wyłącz rejestrowanie audytu.
Rejestruj działania systemowe	Włącz (domyślnie) lub wyłącz rejestrowanie audytu działań związanych z wykrywaniem niskiego poziomu, wykrywaniem sieci oraz autorejestracją wykonywanych przez serwer (użytkownik systemowy).
Włącz wewnętrzny housekeeping	Włącz (domyślnie) lub wyłącz wewnętrzny housekeeping dla rekordów dziennika audytu.
Okres przechowywania danych	Liczba dni, przez które rekordy dziennika audytu powinny być przechowywane, zanim zostaną usunięte przez housekeeper. Wymagane, jeśli housekeeping jest włączony. Domyślnie: 31 dni.

Parametry kompresji dziennika audytu są dostępne w sekcji *Administracja* > *Housekeeping*, w bloku *Kompresja historii, trendów i dziennika audytu*, który staje się widoczny przy użyciu TimescaleDB.

Porządkowanie

Przegląd

Sekcja *Administration* > *Housekeeping* umożliwia skonfigurowanie wewnętrznego housekeeping Zabbix, który automatycznie usuwa z bazy danych nieaktualne lub usunięte przez użytkownika dane. Zapobiega to nieograniczonemu rozrastaniu się bazy danych, co powodowałoby niepotrzebne zużycie miejsca na dysku i pogorszenie wydajności bazy danych.

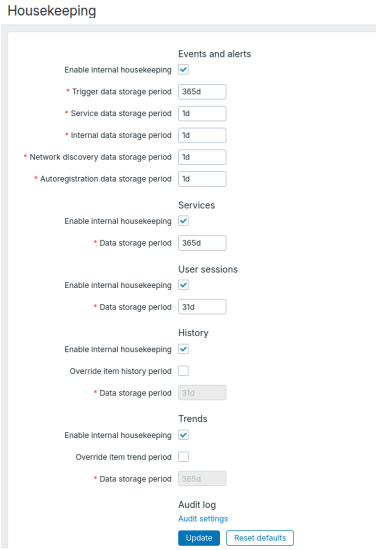
Housekeeping można włączyć i skonfigurować dla następujących typów danych:

- Zdarzenia i alerty z wyzwalaczy, usług, danych wewnętrznych, wykrywania sieci i automatycznej rejestracji
- Usługi
- Sesje użytkowników
- Historia i trendy dla pozycji

Note:
Housekeeping dziennika audytu jest konfigurowany oddzielnie.

Konfiguracja

Dostępne są następujące parametry:



Parameter	Description
Włącz wewnętrzne housekeeping	Włącz lub wyłącz wewnętrzne housekeeping (domyślnie włączone). Jeśli opcja jest włączona, procedura housekeeping automatycznie usuwa z bazy danych dane przekraczające *Okres przechowywania danych*.
Okres przechowywania danych	Określ, jak długo dane mają być przechowywane przed usunięciem przez housekeeping. Wymagane, jeśli wewnętrzne housekeeping jest włączone. Zakres: 1 dzień (1 godzina dla historii) - 25 lat; lub ”0”. Obsługiwane są sufiksy czasu (np. 1d, 1w). Dla *Zdarzeń i alertów* okres przechowywania danych jest ustawiany oddzielnie dla wyzwalaczy, usług, danych wewnętrznych, wykrywania sieci i automatycznej rejestracji. Housekeeper usuwa tylko te zdarzenia, które nie są powiązane z problemami. Na przykład zdarzenie problemu/odzyskania starsze niż *Okres przechowywania danych* nie zostanie usunięte, jeśli jest powiązane z nierozwiązanym problemem. Gdy housekeeper usuwa przestarzałe encje, najpierw usuwa problemy, a następnie zdarzenia. Dla *Historii* i *Trendów* okresy przechowywania danych określają również, jak długo dane pozostają widoczne w *Monitoring* > *Ostatnie dane*, nawet jeśli wewnętrzne housekeeping jest wyłączone. Należy pamiętać, że gdy pozycja lub wyzwalacz zostanie usunięty, jego problemy i powiązane zdarzenia są również usuwane.
Zastąp okres historii pozycji	Jeśli opcja jest włączona, okres przechowywania historii/trendów określony w konfiguracji pozycji zostaje zastąpiony ustawieniem *Okres przechowywania danych* (z wyjątkiem pozycji z włączoną opcją *Nie przechowuj*). Z tej opcji można korzystać nawet wtedy, gdy wewnętrzne housekeeping jest wyłączone i używany jest zewnętrzny housekeeper.
Zastąp okres trendów pozycji	

Jeśli używany jest **ClickHouse**, dostępna staje się sekcja *Okres przechowywania danych ClickHouse*. Wyświetla ona okres prze-

chowywania danych dla każdego typu wartości pozycji. Aby zmienić ten okres, zobacz Konfigurowanie okresu przechowywania danych ClickHouse. Należy pamiętać, że housekeeper nie usuwa danych z ClickHouse.

Jeśli używany jest **Elasticsearch**, dostępna staje się sekcja *Okres przechowywania danych Elasticsearch*. Wyświetla ona typy wartości pozycji zapisywane w Elasticsearch, a okres przechowywania danych jest pokazany jako *Unknown*. Aby kontrolować, jak długo dane są przechowywane, zobacz Index Lifecycle Management. Należy pamiętać, że housekeeper nie usuwa danych z Elasticsearch.

Jeśli używany jest **TimescaleDB**, dostępna staje się sekcja *Kompresja historii, trendów i dziennika audytu*.

Note:
W przypadku TimescaleDB, aby w pełni korzystać z automatycznego partycjonowania, włącz *Zastąp okres historii pozycji*, *Zastąp okres trendów pozycji* oraz *Włącz wewnętrzne housekeeping* dla historii i trendów. Jeśli te opcje są wyłączone, dane przechowywane w tabelach historii i trendów nadal będą partycjonowane, ale housekeeper nie będzie usuwał przestarzałych partycji i pojawią się ostrzeżenia konfiguracyjne. Gdy usuwanie przestarzałych partycji jest włączone, Zabbix server i frontend nie będą już śledzić usuniętych pozycji, a historia dla tych pozycji zostanie wyczyszczona po usunięciu przestarzałej partycji.

Housekeeper jest domyślnie uruchamiany automatycznie co godzinę. Częstotliwość uruchamiania można zmienić, ustawiając parametr konfiguracyjny Zabbix server `HousekeepingFrequency`.

Przycisk *Przywróć domyślne* umożliwia cofnięcie wprowadzonych zmian.

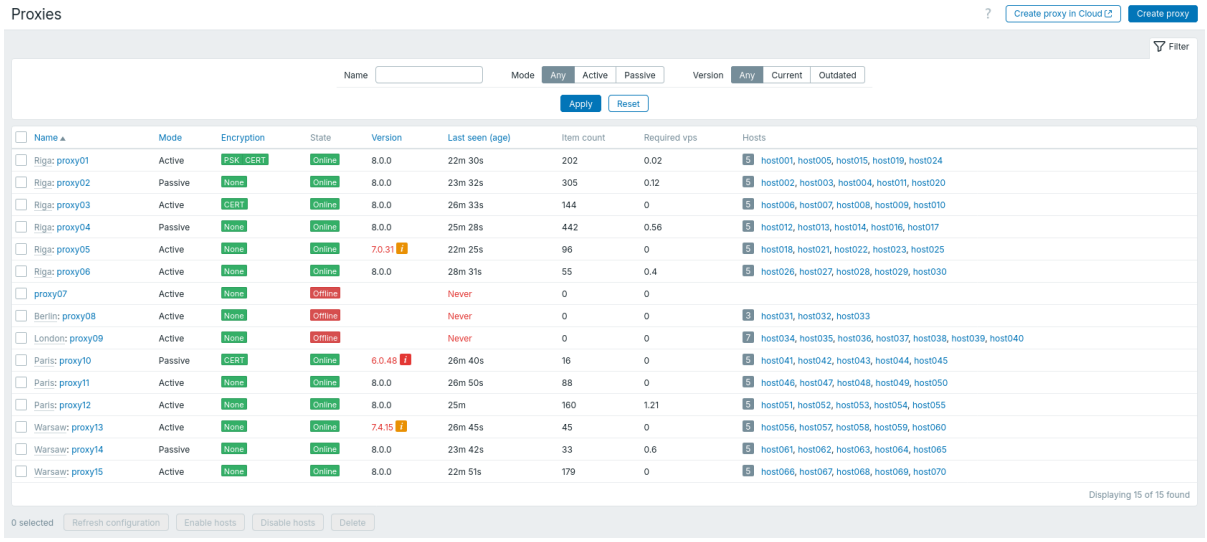
Proxy

Przegląd

W sekcji *Administracja* > *Proxy* w frontend można skonfigurować proxy na potrzeby monitorowania rozproszonego.

Proxy

Wyświetlana jest lista istniejących proxy wraz ze szczegółowymi informacjami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa proxy. Kliknięcie nazwy proxy otwiera formularz konfiguracji proxy. Jeśli proxy należy do grupy proxy, nazwa grupy jest wyświetlana przed nazwą proxy jako szare łącze. Kliknięcie nazwy grupy otwiera formularz konfiguracji grupy proxy.
Tryb	Tryb proxy - *Aktywny* lub *Pasywny*.
Szyfrowanie	Stan szyfrowania połączeń z proxy: **Brak** - brak szyfrowania; **PSK** - używany klucz współdzielony; **Cert** - używany certyfikat.

Kolumna	Opis
Stan	Stan proxy: **Nieznany** - proxy zostało utworzone, gdy serwer Zabbix był wyłączony, lub serwer nie zaktualizował jeszcze stanu; **Online** - proxy komunikowało się z serwerem Zabbix (pasywne proxy odpowiedziało na żądanie serwera; aktywne proxy wysłało żądanie) w okresie przełączania awaryjnego; **Offline** - proxy nie komunikowało się z serwerem Zabbix w okresie przełączania awaryjnego.
Wersja	Wersja proxy (trzycyfrowy numer wersji). Jeśli proxy jest nieaktualne lub nieobsługiwane, numer wersji jest wyróżniony (na czerwono), a ikona stanu informacji (żółta lub czerwona) jest wyświetlana. Umieść wskaźnik myszy nad ikoną, aby wyświetlić szczegóły.
Ostatnio widziane (wiek)	Czas, kiedy serwer ostatnio wykrył proxy.
Liczba pozycji	Liczba włączonych pozycji na włączonych hostach przypisanych do proxy.
Wymagane vps	Wymagana wydajność proxy (liczba wartości, które należy zebrać w ciągu sekundy).
Hosty	Liczba włączonych hostów przypisanych do proxy oraz lista hostów monitorowanych przez proxy. Kliknięcie nazwy hosta otwiera formularz konfiguracji hosta.

Aby skonfigurować nowe proxy, kliknij przycisk *Utwórz proxy* w prawym górnym rogu. Aby wdrożyć nowe proxy w Zabbix Cloud, kliknij znajdujący się obok przycisk *Utwórz proxy w Cloud*. Spowoduje to otwarcie Zabbix Cloud w nowej karcie przeglądarki, w której można utworzyć węzeł proxy. Po utworzeniu węzła proxy skonfiguruj proxy za pomocą przycisku *Utwórz proxy*. Więcej informacji zawiera sekcja Konfigurowanie proxy Zabbix Cloud.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

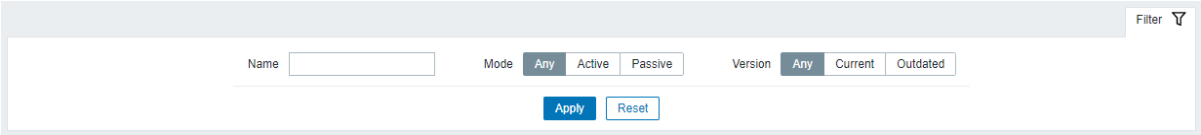
- *Odśwież konfigurację* - odśwież konfigurację proxy;
- *Włącz hosty* - zmień status hostów monitorowanych przez proxy na *Monitorowane*;
- *Wyłącz hosty* - zmień status hostów monitorowanych przez proxy na *Nemonitorowane*;
- *Usuń* - usuń proxy.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi proxy, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko interesujące Cię proxy. Aby zwiększyć wydajność wyszukiwania, dane są wyszukiwane z nierozwiniętymi makrami.

Link *Filtr* jest dostępny nad listą proxy. Po jego kliknięciu dostępny jest filtr, za pomocą którego można filtrować proxy według nazwy, trybu i wersji. Należy pamiętać, że opcja filtra *Nieaktualne* wyświetla zarówno nieaktualne (częściowo obsługiwane), jak i nieobsługiwane proxy.



Grupy proxy

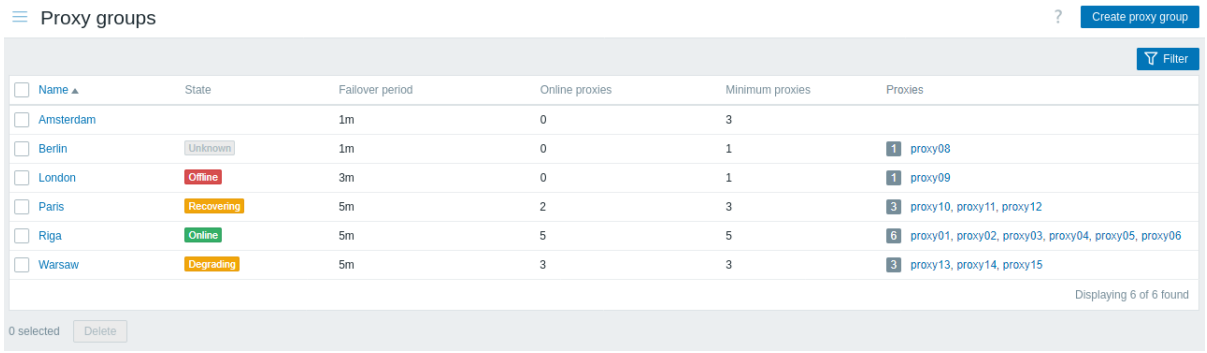
Przegląd

W *Administration* > *Proxy groups* można skonfigurować grupy proxy.

Grupy proxy są używane w równoważeniu obciążenia proxy z automatycznym rozdzielaniem hostów między proxy oraz wysoką dostępnością między proxy.

Grupy proxy

Wyświetlana jest lista istniejących grup proxy wraz ze szczegółowymi informacjami.



Wyświetlane dane:

Kolumna	Opis
Nazwa	Nazwa grupy proxy. Kliknięcie nazwy grupy proxy otwiera formularz konfiguracji grupy proxy.
Stan	Stan grupy proxy (wyświetlany, jeśli grupa zawiera co najmniej jeden proxy): **Nieznany** - grupa została utworzona, gdy serwer Zabbix był wyłączony, lub serwer nie zaktualizował jeszcze stanu; **Online** - minimalna liczba proxy skomunikowała się z serwerem Zabbix w okresie przełączania awaryjnego; **Pogarszający się** - grupa wkrótce przejdzie w stan offline, ponieważ liczba proxy online spadła poniżej wartości progowej *Minimalna liczba proxy*; **Offline** - mniej niż minimalna liczba proxy skomunikowała się z serwerem Zabbix w okresie przełączania awaryjnego; **Odzyskiwanie** - grupa wkrótce powróci do stanu online, ponieważ liczba proxy online osiągnęła wartość progową *Minimalna liczba proxy*. Stan grupy proxy można monitorować za pomocą pozycji zabbix[proxy group,<name>,state] i zabbix[proxy group,discovery].
Okres przełączania awaryjnego	Okres, w którym proxy w grupie proxy musi skomunikować się z serwerem Zabbix, aby zostać uznane za online.
Proxy online	Liczba proxy online (wyświetlana na czerwono, jeśli jest niższa od minimalnej liczby określonej dla grupy).
Minimalna liczba proxy	Minimalna liczba proxy online wymagana do utrzymania grupy proxy w stanie online.
Proxy	Liczba proxy w grupie oraz lista proxy w grupie wraz z odnośnikami do formularza konfiguracji proxy. Maksymalna liczba wyświetlanych proxy jest ograniczona wartością *Maksymalna liczba elementów wyświetlanych w komórce tabeli*.

Aby skonfigurować nową grupę proxy, kliknij przycisk *Utwórz grupy proxy* w prawym górnym rogu.

Opcje masowej edycji

Przyciski poniżej listy oferują kilka opcji masowej edycji:

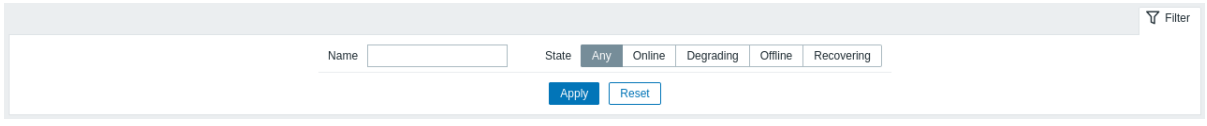
- *Usuń* - usuń grupy proxy.

Aby użyć tych opcji, zaznacz pola wyboru przed odpowiednimi grupami proxy, a następnie kliknij wymagany przycisk.

Korzystanie z filtra

Możesz użyć filtra, aby wyświetlić tylko te grupy proxy, które Cię interesują. Aby zwiększyć wydajność wyszukiwania, dane są przeszukiwane z nierozwiązanymi makrami.

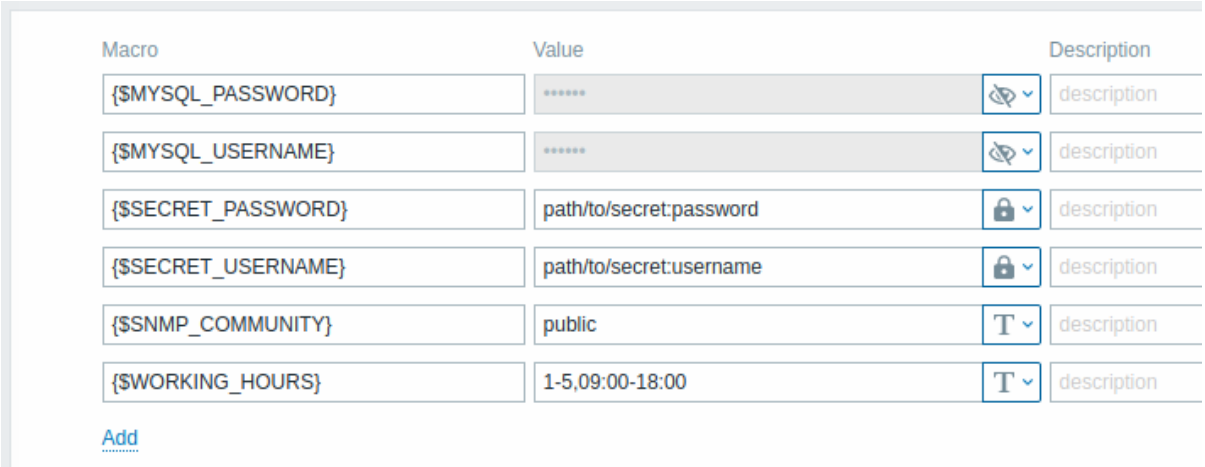
Łącze *Filter* jest dostępne nad listą grup proxy. Po jego kliknięciu pojawia się filtr, za pomocą którego można filtrować grupy proxy według nazwy i statusu.



Makra

Przegląd

W tej sekcji można definiować makra użytkownika obowiązujące w całym systemie jako pary nazwa-wartość. Należy pamiętać, że wartości makr mogą być przechowywane jako zwykły tekst, tekst poufny lub sekret Vault. Można również dodać opis.



Kolejka

Przegląd

W sekcji *Administracja* > *Kolejka* wyświetlane są pozycje oczekujące na aktualizację.

Idealnie po otwarciu tej sekcji wszystko powinno być „zielone”, co oznacza brak pozycji w kolejce. Jeśli wszystkie pozycje są aktualizowane bez opóźnień, żadna z nich nie oczekuje. Jednak z powodu niewystarczającej wydajności serwera aktualizacja niektórych pozycji może się opóźniać, a informacje o tym są wyświetlane w tej sekcji. Więcej informacji można znaleźć w sekcji Kolejka.

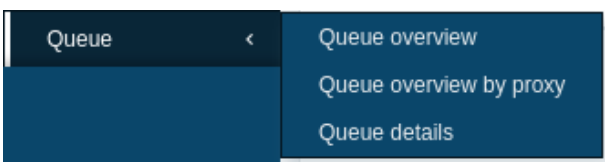
Note:
proxy Zabbix nie uwzględnia okresów konserwacji; szczegółowe informacje można znaleźć w sekcji Obliczanie kolejek podczas konserwacji.

Note:
Kolejka jest dostępna tylko wtedy, gdy działa serwer Zabbix. Pozycje nie są uwzględniane w kolejce, jeśli interfejs pozycji stanie się niedostępny z powodu problemów z połączeniem lub nieprawidłowego działania agenta.

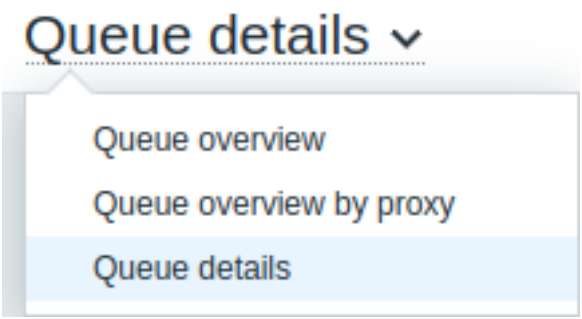
Sekcja *Administracja* > *Kolejka* zawiera następujące strony:

- Przegląd kolejki - wyświetla kolejkę według typu pozycji;
- Przegląd kolejki według proxy - wyświetla kolejkę według proxy;
- Szczegóły kolejki - wyświetla listę opóźnionych pozycji.

Lista dostępnych stron pojawia się po kliknięciu pozycji *Kolejka* w sekcji menu *Administracja*. Można również przełączać się między stronami za pomocą listy rozwijanej tytułu w lewym górnym rogu.



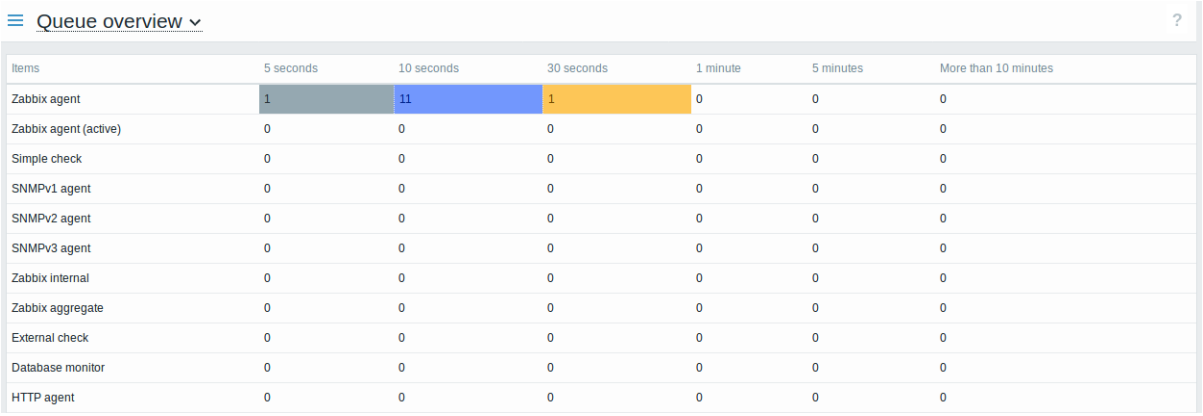
Menu trzeciego poziomu.



Lista rozwijana tytułu.

Przegląd według typu pozycji

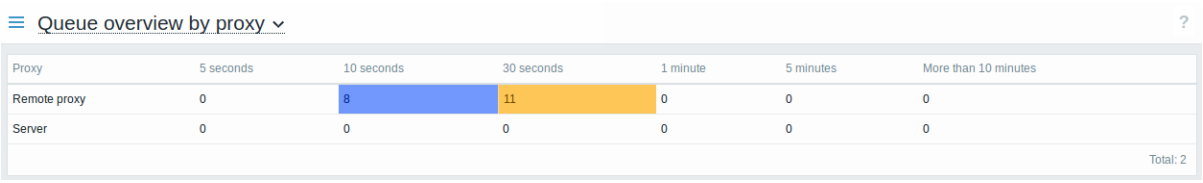
Na tym ekranie łatwo ustalić, czy problem jest związany z jednym czy z kilkoma typami pozycji.



Każdy wiersz zawiera typ pozycji. Każda kolumna pokazuje liczbę oczekujących pozycji — oczekujących odpowiednio przez 5–10 sekund / 10–30 sekund / 30–60 sekund / 1–5 minut / 5–10 minut lub ponad 10 minut.

Przegląd według proxy

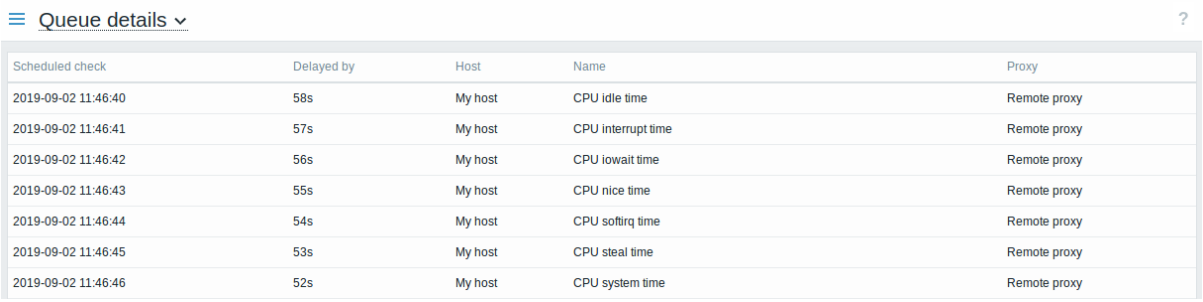
Na tym ekranie łatwo ustalić, czy problem jest związany z jednym z proxy czy z serwerem.



Każdy wiersz zawiera proxy, a na końcu listy znajduje się serwer. Każda kolumna pokazuje liczbę oczekujących pozycji — oczekujących odpowiednio przez 5–10 sekund / 10–30 sekund / 30–60 sekund / 1–5 minut / 5–10 minut lub ponad 10 minut.

Lista oczekujących pozycji

Na tym ekranie wyświetlana jest każda oczekująca pozycja.



Wyświetlane dane:

Kolumna	Opis
Zaplanowane sprawdzenie	Wyświetlany jest czas, w którym sprawdzenie powinno zostać wykonane.
Opóźnienie	Wyświetlana jest długość opóźnienia.
Host	Wyświetlany jest host pozycji.
Nazwa	Wyświetlana jest nazwa oczekującej pozycji.
Proxy	Wyświetlana jest nazwa proxy, jeśli host jest monitorowany przez proxy.

Możliwe komunikaty o błędach

Może wystąpić sytuacja, w której nie są wyświetlane żadne dane i pojawia się następujący komunikat o błędzie:



W tym przypadku komunikat o błędzie jest następujący:

```
Cannot display item queue. Permission denied
```

Dzieje się tak, gdy parametry konfiguracji PHP w pliku konfiguracyjnym frontendu - `$ZBX_SERVER` lub oba `$ZBX_SERVER` i `$ZBX_SERVER_PORT` - wskazują istniejący serwer Zabbix, który korzysta z innej bazy danych.

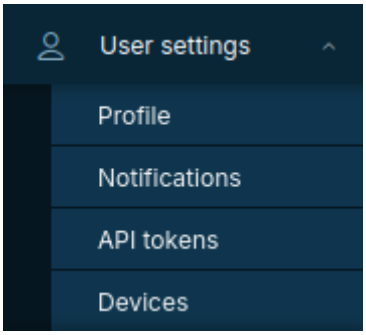
Ustawienia użytkownika

Przegląd

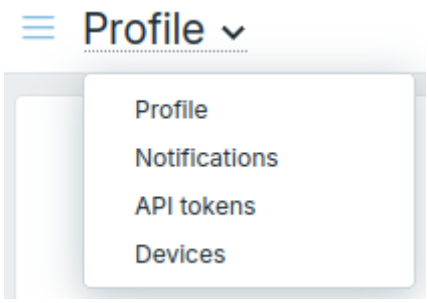
W zależności od roli użytkownika sekcja *Ustawienia użytkownika* może zawierać następujące strony:

- *Profil* lub *Profil użytkownika* - dostosowywanie niektórych funkcji frontend Zabbix;
- *Powiadomienia* - dostosowywanie powiadomień bieżącego użytkownika;
- *Tokeny API* - zarządzanie tokenami API przypisanymi do bieżącego użytkownika;
- *Urządzenia* - zarządzanie urządzeniami mobilnymi należącymi do bieżącego użytkownika.

Lista dostępnych stron pojawia się po kliknięciu ikony użytkownika w pobliżu dolnej części menu Zabbix (niedostępna dla użytkownika *guest*). Możliwe jest również przełączanie między stronami za pomocą listy rozwijanej tytułu w lewym górnym rogu.



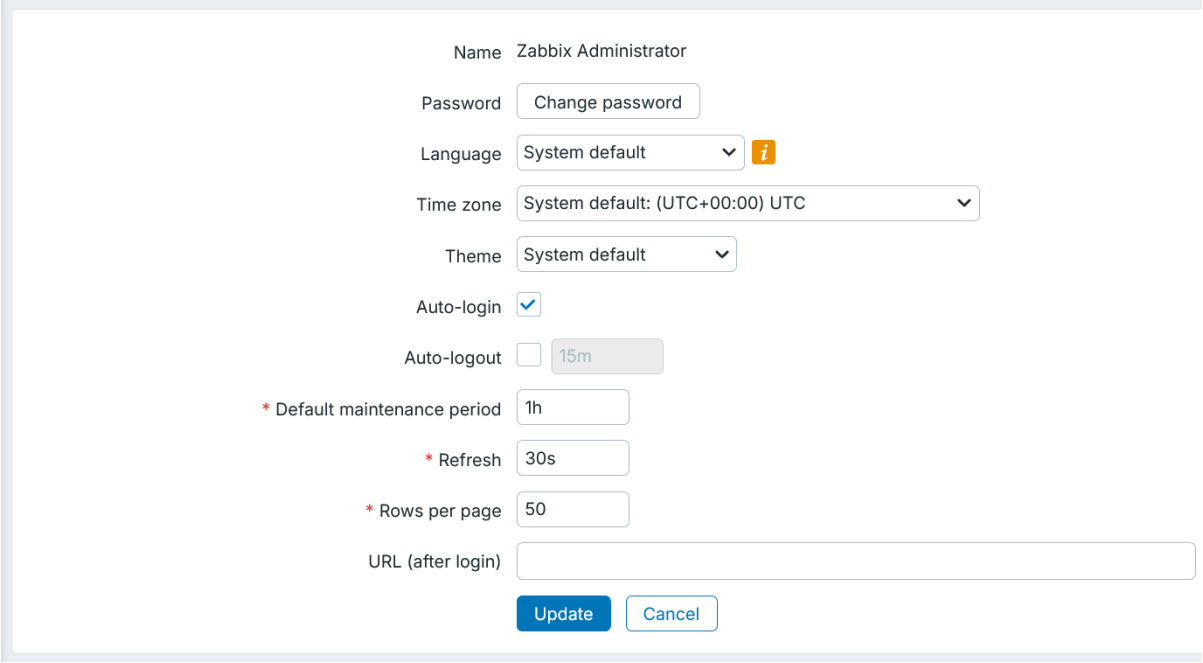
Menu trzeciego poziomu.



Lista rozwijana tytułu.

Profil użytkownika

Sekcja *Profil użytkownika* zawiera opcje ustawienia niestandardowego języka interfejsu, motywu kolorystycznego, liczby wierszy wyświetlanych na listach itp. Wprowadzone tutaj zmiany zostaną zastosowane tylko do bieżącego użytkownika.



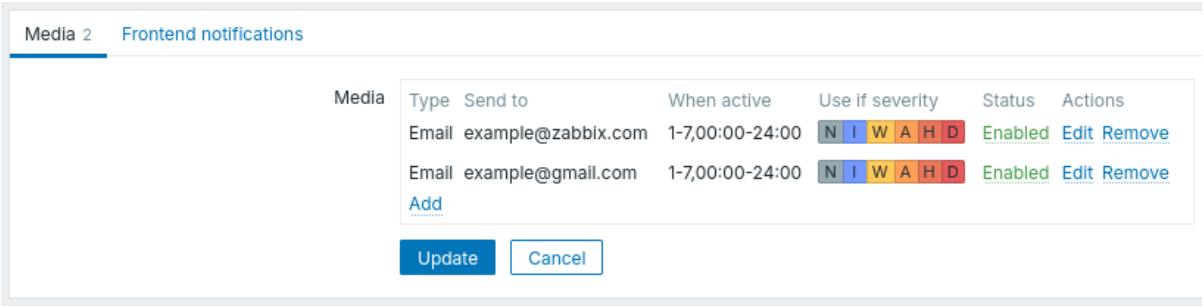
Parametr	Opis
Nazwa	Wyświetlana jest nazwa i nazwisko użytkownika. Jeśli nazwa i nazwisko nie są podane, wyświetlana jest nazwa użytkownika.

Parametr	Opis
Hasło	Kliknij przycisk *Zmień hasło*, aby otworzyć trzy pola: *Stare hasło*, *Nowe hasło*, *Nowe hasło (ponownie)*. Po pomyślnej zmianie hasła użytkownik zostanie wylogowany ze wszystkich aktywnych sesji. Pamiętaj, że hasło można zmienić tylko użytkownikom korzystającym z uwierzytelniania wewnętrznego Zabbix.
Język	Wybierz preferowany język interfejsu lub wybierz **Domyślne ustawienia systemu**, aby użyć domyślnych ustawień systemowych. Wybranie *English (en_US)* włączy również amerykański format godziny i daty w frontendzie. Więcej informacji można znaleźć w sekcji Instalowanie dodatkowych języków frontendu.
Strefa czasowa	Wybierz strefę czasową, aby zastąpić globalną strefę czasową na poziomie użytkownika, lub wybierz **Domyślne ustawienia systemu**, aby użyć globalnych ustawień strefy czasowej.
Motyw	Wybierz motyw kolorystyczny dla swojego profilu: **Domyślne ustawienia systemu** - używa domyślnych ustawień systemowych; **Niebieski** - standardowy niebieski motyw; **Niebieski (klasyczny)** - standardowy niebieski motyw z użyciem starszych czcionek; **Ciemny** - alternatywny ciemny motyw; **Ciemnoniebieski** - ciemnoniebieski motyw; **Ciemny (klasyczny)** - alternatywny ciemny motyw z użyciem starszych czcionek; **Jasny o wysokim kontraście** - jasny motyw o wysokim kontraście; **Ciemny o wysokim kontraście** - ciemny motyw o wysokim kontraście.
Automatyczne logowanie	Zaznacz to pole wyboru, aby Zabbix zapamiętał użytkownika i automatycznie logował go przez 30 dni. Podczas logowania z włączoną opcją *Zapamiętaj przez 30 dni*: - Licznik automatycznego wylogowania jest resetowany (sesja trwa do końca 30-dniowego okresu). - Automatyczne logowanie jest włączane podczas kolejnych wizyt w ciągu 30 dni. Podczas logowania bez opcji *Zapamiętaj przez 30 dni*: - Parametr automatycznego logowania jest czyszczony. - Automatyczne wylogowanie działa jak zwykle, zgodnie ze skonfigurowanym limitem czasu. Do tego celu używane są pliki cookie przeglądarki.
Automatyczne wylogowanie	Po zaznaczeniu tego pola wyboru użytkownik zostanie automatycznie wylogowany po upływie ustawionej liczby sekund (minimum 90 sekund, maksimum 1 dzień). Obsługiwane są przyrostki czasu, na przykład: 90s, 5m, 2h, 1d. Pamiętaj, że ta opcja nie będzie działać w następujących przypadkach: * Gdy strony menu *Monitorowanie* wykonują odświeżanie informacji w tle. Jeśli strony odświeżające dane w określonych odstępach czasu (panele, wykresy, najnowsze dane itp.) pozostaną otwarte, czas trwania sesji zostanie odpowiednio wydłużony, co wyłączy funkcję automatycznego wylogowania. * Jeśli logowanie odbywa się z zaznaczoną opcją *Zapamiętaj mnie przez 30 dni*; w takim przypadku licznik automatycznego wylogowania jest zastępowany, a sesja pozostaje aktywna przez cały okres. * Gdy uwierzytelnianie odbywa się za pośrednictwem dostawcy tożsamości SSO (IdP), automatyczne wylogowanie kończy tylko sesję Zabbix. Sesja SSO z IdP pozostaje aktywna, więc użytkownik może móc zalogować się ponownie bez podawania hasła. *Automatyczne wylogowanie* może również przyjmować wartość „0”, co oznacza wyłączenie funkcji automatycznego wylogowania po zaktualizowaniu ustawień profilu.
Domyślny okres konserwacji	Ustaw domyślny okres konserwacji dla konserwacji ad hoc.
Odświeżanie	Ustaw częstotliwość odświeżania informacji na stronach menu *Monitorowanie* (minimum 0 sekund, maksimum 1 godzina). Obsługiwane są przyrostki czasu, na przykład: 30s, 90s, 1m, 1h.
Wierszy na stronę	Ustaw liczbę wierszy wyświetlanych na stronie list. Mniejsza liczba wierszy (i mniej rekordów do wyświetlenia) skutkuje krótszym czasem ładowania.
URL (po zalogowaniu)	Ustaw konkretny URL, który będzie wyświetlany po zalogowaniu. Zamiast domyślnych *Paneli* może to być na przykład URL strony *Monitorowanie* > *Wyzwalacze*. To pole jest wyłączone, jeśli użytkownik *Super admin* ograniczył URL po zalogowaniu dla danej roli użytkownika do określonej strony. Jeśli pole jest wyłączone lub użytkownik nie określi żadnej wartości, poniżej pola wyświetlana jest wartość domyślna, np. `zabbix.php?action=dashboard.view`

Powiadomienia

Sekcja *Powiadomienia* udostępnia opcje dostosowywania powiadomień bieżącego użytkownika.

Karta *Media* umożliwia określenie szczegółów mediów dla użytkownika, takich jak typy mediów i adresy, których należy używać, oraz kiedy należy ich używać do dostarczania powiadomień.



Jeśli typ mediów został wyłączony:

- po nazwie wyświetlana jest żółta ikona informacyjna;
- w kolumnie *Status* wyświetlane jest "Disabled".

Note:
Uprawnienia użytkownika do zmiany własnych szczegółów mediów mogą być przyznawane/odbierane na podstawie jego roli użytkownika (zobacz opcję *Create and edit own media*). Uprawnienia użytkownika Super admin do zmiany szczegółów mediów innych użytkowników również mogą być przyznawane/odbierane na podstawie jego roli użytkownika (zobacz opcję *Create and edit user media*)

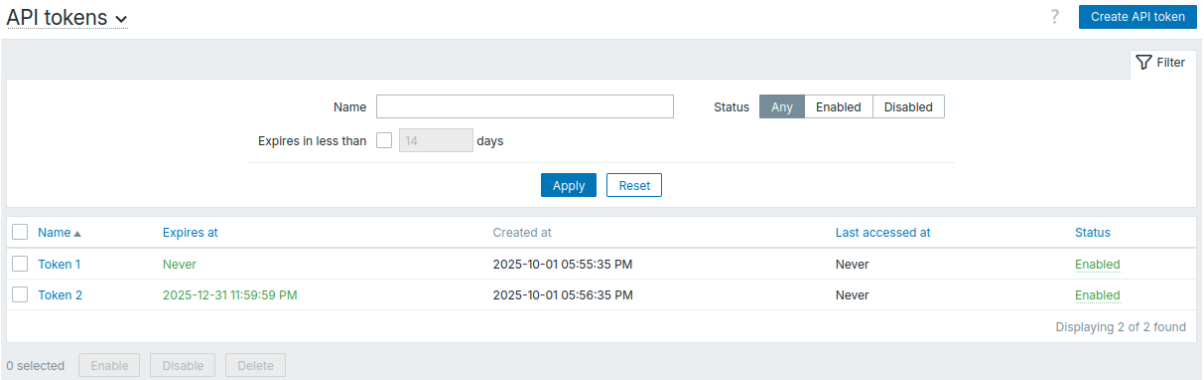
Należy pamiętać, że w przypadku użytkowników provisioned:

- media użytkownika provisioned nie mogą zostać usunięte;
- media użytkownika provisioned mogą być wyłączane/włączane;
- takie pola mediów użytkownika provisioned jak *When active*, *Use if severity* i *Enabled* mogą być edytowane ręcznie;
- dodatkowe media użytkownika mogą być dodawane ręcznie dla użytkowników provisioned (na przykład dodatkowe adresy e-mail);
- media użytkownika dodane ręcznie mogą zostać usunięte.

Karta *Frontend notifications* umożliwia ustawienie powiadomień globalnych.

Tokeny API

Sekcja *Tokeny API* umożliwia wyświetlanie tokenów przypisanych do użytkownika, edycję szczegółów tokenu oraz tworzenie nowych tokenów. Ta sekcja jest dostępna dla użytkownika tylko wtedy, gdy akcja *Zarządzanie tokenami API* jest dozwolona w ustawieniach roli użytkownika.



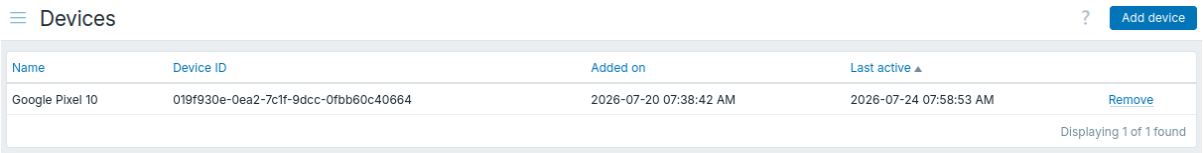
Możesz filtrować tokeny API według nazwy, daty wygaśnięcia lub statusu (*Włączony/Wyłączony*). Kliknij status tokenu na liście, aby szybko włączyć/wyłączyć token. Możesz również włączyć/wyłączyć wiele tokenów jednocześnie, zaznaczając je na liście, a następnie klikając przyciski *Włącz/Wyłącz* poniżej listy.

Attention:
Użytkownicy nie mogą wyświetlić wartości *Auth token* tokenów przypisanych do nich w Zabbix. Wartość *Auth token* jest wyświetlana tylko raz - bezpośrednio po utworzeniu tokenu. Jeśli token został utracony, należy go wygenerować ponownie.

Urządzenia

Sekcja *Urządzenia* umożliwia dodawanie urządzeń mobilnych używanych z aplikacją mobilną Zabbix oraz ich usuwanie.

Ta sekcja jest wyświetlana, jeśli włączono `EnableMobileDevices`.



Aby dodać urządzenie:

1. Kliknij *Dodaj urządzenie*.
2. Zeskanuj kod QR w aplikacji mobilnej Zabbix.

Note:
Po dodaniu urządzenia Zabbix dodaje również typ nośnika powiadomień push do nośników użytkownika. Po usunięciu urządzenia ten nośnik użytkownika również zostaje usunięty. Jeśli skonfigurowano wiele typów nośników powiadomień push, Zabbix dodaje ten, który jest włączony i znajduje się pierwszy w kolejności alfabetycznej.

Dodane urządzenia są wyświetlane na liście. Lista zawiera ich nazwy i identyfikatory, a także daty i godziny dodania oraz ostatniej aktywności.

Można również usuwać urządzenia. Po usunięciu urządzenia jego dane uwierzytelniające dostępu zostają unieważnione i usunięte z Zabbix, po czym urządzenie nie może już uzyskać dostępu do Zabbix.

Superadministratorzy mogą również zarządzać urządzeniami innych użytkowników w sekcji *Użytkownicy* > *Urządzenia*.

Powiadomienia globalne

Przegląd

Powiadomienia globalne umożliwiają wyświetlanie problemów w czasie rzeczywistym bezpośrednio na bieżącym ekranie w frontend Zabbix.

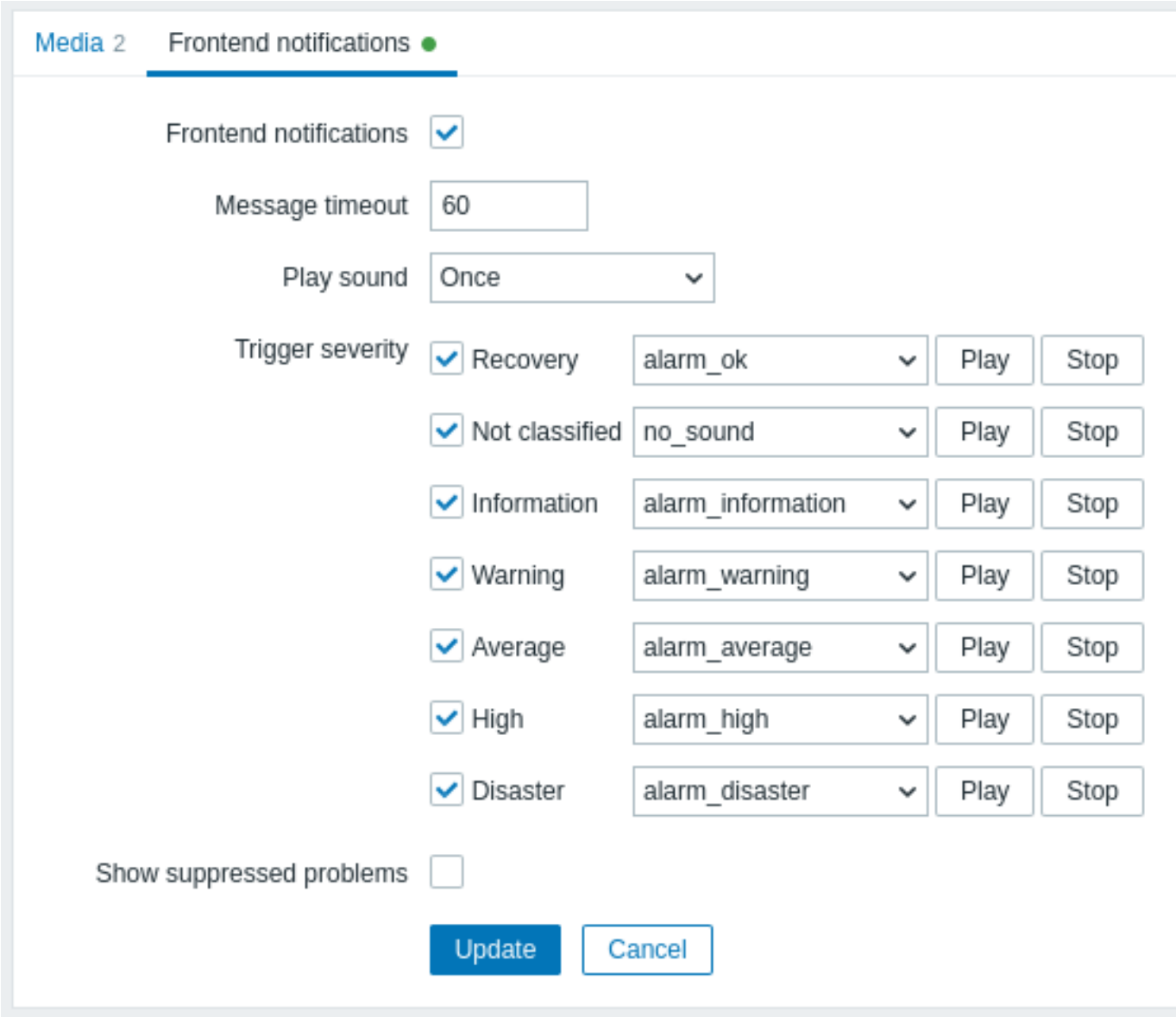
Bez powiadomień globalnych, podczas pracy poza sekcjami *Problemy* lub *Pulpit*, nie otrzymywałbyś żadnych informacji o bieżących problemach. Powiadomienia globalne zapewniają wyświetlanie tych informacji niezależnie od bieżącej lokalizacji w frontend Zabbix.

Powiadomienia globalne obejmują zarówno wyświetlanie komunikatu, jak i odtwarzanie dźwięku.

Attention:
Automatyczne odtwarzanie dźwięków może być wyłączone (domyślnie) w najnowszych wersjach przeglądarek. W takich przypadkach należy włączyć to ustawienie ręcznie.

Konfiguracja

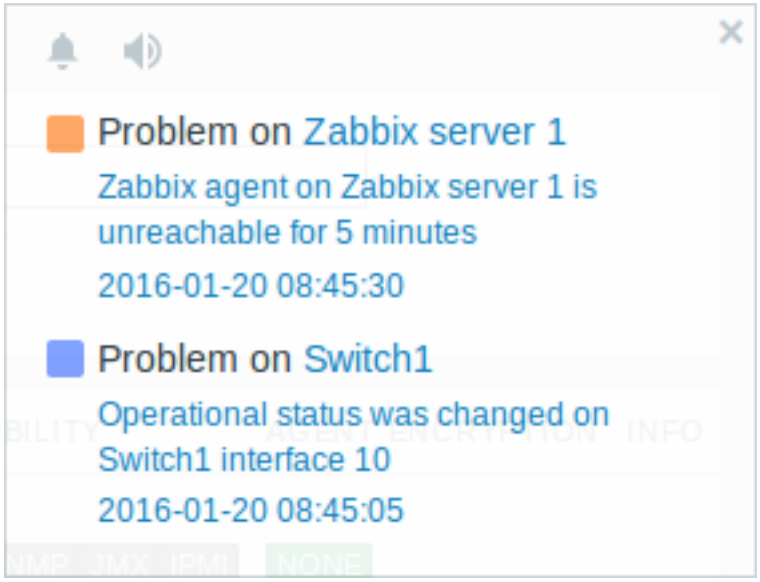
Globalne powiadomienia można włączyć dla każdego użytkownika na karcie **Powiadomienia frontend** w sekcji Powiadomienia.



Parametr	Opis
Powiadomienia frontend	Zaznacz pole wyboru, aby włączyć globalne powiadomienia.
Limit czasu wiadomości	Ustaw czas, przez jaki wiadomość będzie wyświetlana. Domyślnie wiadomości pozostają na ekranie przez 60 sekund. Obsługiwane są sufiksy czasu, na przykład: 30s, 5m, 2h, 1d.
Odtwarzaj dźwięk	Ustaw czas, przez jaki dźwięk będzie odtwarzany. **Raz** - dźwięk jest odtwarzany raz w całości; **10 sekund** - dźwięk jest powtarzany przez 10 sekund; **Limit czasu wiadomości** - dźwięk jest powtarzany, dopóki wiadomość jest widoczna.
Poziom ważności wyzwalacza	Ustaw poziomy ważności wyzwalaczy, dla których będą aktywowane globalne powiadomienia i dźwięki. Możesz także wybrać dźwięki odpowiednie dla różnych poziomów ważności. Jeśli nie zostanie zaznaczony żaden poziom ważności, żadne wiadomości nie będą wyświetlane. Dodatkowo wiadomości o odzyskaniu będą wyświetlane tylko dla zaznaczonych poziomów ważności. Na przykład, jeśli zaznaczone są *Odzyskanie* i *Katastrofa*, globalne powiadomienia będą wyświetlane dla problemów i odzyskań wyzwalaczy o poziomie ważności *Katastrofa*.
Pokaż stłumione problemy	Zaznacz pole wyboru, aby wyświetlać powiadomienia o problemach, które w przeciwnym razie byłyby stłumione (niepokazywane) z powodu konserwacji hosta.

Wyświetlane komunikaty globalne

Gdy komunikaty napływają, są wyświetlane w pływającej sekcji po prawej stronie. Możesz dowolnie zmieniać położenie tej sekcji, przeciągając jej nagłówek.



Dla tej sekcji dostępnych jest kilka elementów sterujących:

- Przycisk **Snooze** wycisza aktualnie aktywny dźwięk alarmu;
- Przycisk **Mute/Unmute** przełącza między odtwarzaniem a całkowitym nieodtwarzaniem dźwięków alarmu.

Dźwięk w przeglądarkach

Przegląd

Dźwięk jest używany w powiadomieniach globalnych.

Aby dźwięki były odtwarzane w frontend Zabbix, w profilu użytkownika na karcie *Frontend notifications* musi być włączona opcja *Frontend notifications*, a wszystkie poziomy ważności wyzwalaczy muszą być zaznaczone. Dodatkowo dźwięki powinny być włączone w wyskakującym oknie powiadomień globalnych.

Jeśli z jakiegokolwiek powodu dźwięk nie może być odtwarzany na urządzeniu, przycisk w wyskakującym oknie powiadomień globalnych pozostanie na stałe w stanie „wyciszenia", a po najechaniu kursorem na przycisk zostanie wyświetlony komunikat „Cannot support notification audio for this device".

Dźwięki, w tym domyślne klipy audio, są obsługiwane wyłącznie w formacie MP3.

Dźwięki frontend Zabbix zostały pomyślnie przetestowane w najnowszych przeglądarkach Firefox i Opera w systemie Linux oraz w przeglądarkach Chrome, Firefox, Microsoft Edge i Opera w systemie Windows.

Attention:
Automatyczne odtwarzanie dźwięków może być wyłączone (domyślnie) w najnowszych wersjach przeglądarek. W takich przypadkach należy włączyć to ustawienie ręcznie.

Globalne wyszukiwanie

W frontendzie Zabbix można wyszukiwać hosty, grupy hostów, szablony i grupy szablonów.

Pole wyszukiwania znajduje się pod logo Zabbix w menu. Wyszukiwanie można uruchomić, naciskając *Enter* lub klikając ikonę wyszukiwania .



Jeśli istnieje host, który zawiera wprowadzony ciąg znaków w dowolnej części nazwy, zostanie wyświetlona lista rozwijana zawierająca wszystkie takie hosty (z dopasowaną częścią wyróżnioną na pomarańczowo). Lista rozwijana wyświetli również host, jeśli jego widoczna nazwa odpowiada technicznej nazwie wpisanej jako ciąg wyszukiwania; dopasowany host zostanie wyświetlony, ale bez wyróżnienia.

Atrybuty podlegające wyszukiwaniu

Hosty można wyszukiwać według następujących właściwości:

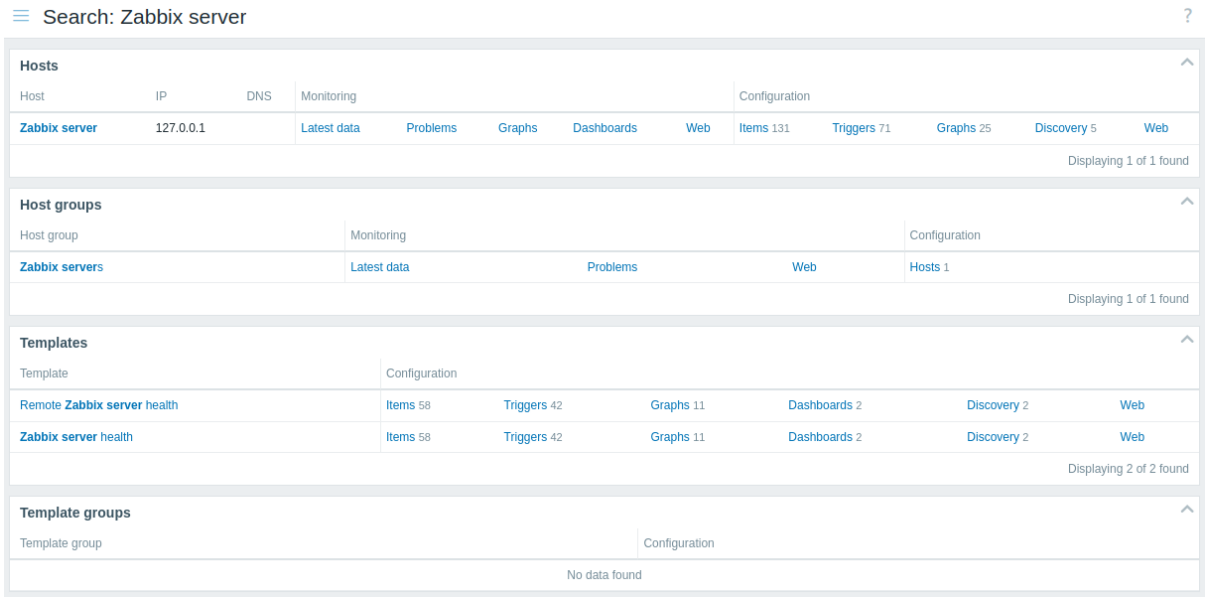
- Nazwa hosta
- Widoczna nazwa
- Adres IP
- Nazwa DNS

Szablony można wyszukiwać według nazwy lub widocznej nazwy. Jeśli wyszukujesz według nazwy, która różni się od widocznej nazwy (szablonu/hosta), w wynikach wyszukiwania jest ona wyświetlana poniżej widocznej nazwy w nawiasach.

Grupy hostów i szablonów można wyszukiwać według nazwy. Określenie grupy nadrzędnej niejawnie wybiera wszystkie grupy zagnieżdżone.

Wyniki wyszukiwania

Wyniki wyszukiwania składają się z czterech oddzielnych bloków dla hostów, grup hostów, szablonów i grup szablonów.



Możliwe jest zwijanie/rozwijanie każdego pojedynczego bloku. Liczba wpisów jest wyświetlana na dole każdego bloku, na przykład: *Wyświetlanie 13 z 13 znalezionych*. Jeśli nie ma żadnych wpisów, liczba wpisów nie jest wyświetlana. Łączna liczba wpisów wyświetlanych w jednym bloku jest ograniczona do 100.

Każdy wpis zawiera odnośniki do danych monitoringu i konfiguracji. Zobacz pełną listę odnośników.

Dla wszystkich danych konfiguracyjnych (takich jak pozycje, wyzwalacze, wykresy) liczba znalezionych encji jest wyświetlana jako liczba obok nazwy encji, w kolorze szarym. **Uwaga:** jeśli encji jest zero, żadna liczba nie jest wyświetlana.

Włączone hosty są wyświetlane na niebiesko, wyłączone hosty na czerwono.

Dostępne linki

Dla każdego wpisu dostępne są następujące linki:

- Hosty
  - Monitorowanie
    - Najnowsze dane
    - Problemy
    - Wykresy
    - Pulpity hosta
    - Scenariusze WWW
  - Konfiguracja
    - Pozycje
    - Wyzwalacze
    - Wykresy
    - Reguły wykrywania
    - Scenariusze WWW
- Grupy hostów
  - Monitorowanie
    - Najnowsze dane
    - Problemy
    - Scenariusze WWW
  - Konfiguracja
    - Hosty
- Szablony
  - Konfiguracja
    - Pozycje
    - Wyzwalacze
    - Wykresy
    - Pulpity szablonu
    - Reguły wykrywania
    - Scenariusze WWW
- Grupy szablonów
  - Konfiguracja
    - Szablony

Tryb konserwacji frontend

Przegląd

Można tymczasowo wyłączyć frontend Zabbix, aby ograniczyć dostęp. Służy do ochrony bazy danych Zabbix przed zmianami wprowadzonymi przez użytkownika, zachowując jej integralność.

Gdy frontend Zabbix jest w trybie konserwacji, możesz bezpiecznie zatrzymać bazę danych i wykonać zadania konserwacyjne.

Użytkownicy ze zdefiniowanych adresów IP będą mogli normalnie wchodzić w interakcję z frontendem w trybie konserwacji.

Konfiguracja

Aby włączyć tryb konserwacji, otwórz plik `maintenance.inc.php` (znajdujący się w katalogu `/conf` katalogu dokumentów HTML Zabbix na serwerze WWW) i usuń komentarze z następujących wierszy:

```
// Tryb konserwacji.
define('ZBX_DENY_GUI_ACCESS', 1);

// Tablica adresów IP, z których można łączyć się z frontendem (opcjonalnie).
$ZBX_GUI_ACCESS_IP_RANGE = array('127.0.0.1');

// Komunikat wyświetlany na ekranie ostrzeżenia (opcjonalnie).
$ZBX_GUI_ACCESS_MESSAGE = 'Trwa aktualizacja bazy danych MySQL do godziny 15:00. Prosimy czekać...';
```

Note:
W większości przypadków plik `maintenance.inc.php` znajduje się w katalogu `/conf` katalogu dokumentów HTML Zabbix na serwerze WWW. Jednak niektóre systemy operacyjne i serwery WWW mogą używać innej lokalizacji.
Na przykład lokalizacja dla:

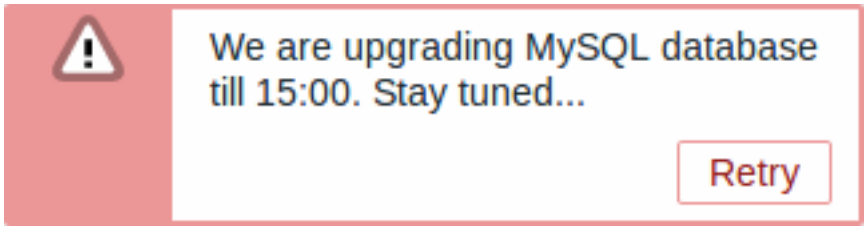
- SUSE i RedHat to `/etc/zabbix/web/maintenance.inc.php`.
- systemów opartych na Debianie to `/usr/share/zabbix/conf/`.

Zobacz także Kopiowanie plików PHP.

Parametr	Szczegóły
ZBX_DENY_GUI_ACCESS	Jeśli parametr zostanie zdefiniowany z dowolną wartością, tryb konserwacji zostanie włączony. Aby wyłączyć tryb konserwacji, zakomentuj parametr lub usuń go.
ZBX_GUI_ACCESS_IP_RANGE	Tablica adresów IP, z których można łączyć się z frontendem (opcjonalnie). Na przykład: `array('192.168.1.1', '192.168.1.2')`
ZBX_GUI_ACCESS_MESSAGE	Komunikat informujący użytkowników o konserwacji (opcjonalnie). Jeśli parametr nie zostanie zdefiniowany, zostanie użyty domyślny komunikat *„Zabbix jest w trakcie konserwacji”*.

Wyświetl

Użytkownicy zobaczą następujący ekran podczas próby dostępu do frontendu Zabbix w trybie konserwacji. Ekran jest odświeżany co 30 sekund, aby powrócić do normalnego stanu bez interwencji użytkownika po zakończeniu konserwacji.



Adresy IP zdefiniowane w *ZBX_GUI_ACCESS_IP_RANGE* będą mogły uzyskać dostęp do frontendu jak zwykle.

Parametry strony

Przegląd

Większość stron interfejsu webowego Zabbix obsługuje różne parametry HTTP GET, które określają, co zostanie wyświetlone. Można je przekazywać, podając po adresie URL pary parametr=wartość, oddzielone od adresu URL znakiem zapytania (?) i od siebie nawzajem znakiem ampersand (&).

Monitoring > Problemy

Obsługiwane są następujące parametry:

Parameter	Description	Example
show	Opcja filtra *Show*. Possible values: 1 - recent problems; 2 - all; 3 - in problem state.	show=1
name	Opcja filtra *Problem*: dowolny ciąg znaków.	name=Zabbix agent

Parameter	Description	Example
severities	Opcja filtra *Severity*: tablica wybranych poziomów ważności w formacie `severities[*]=*` (zastąp `*` poziomem ważności). Possible values: 0 - not classified; 1 - information; 2 - warning; 3 - average; 4 - high; 5 - disaster.	severities[3]=3
inventory	Opcja filtra *Host inventory*: tablica pól inwentarza `[field]`,`[value]`	inventory[0][field]=type&invento
evaltype	Opcja filtra *Tags*: metoda oceny tagów. Possible values: 0 - and/or; 2 - or.	evaltype=0
tags	Opcja filtra *Tags*: tablica zdefiniowanych tagów `[tag]`, `[operator]`, `[value]` Possible operator values: 0 - contains; 1 - equals; 2 - does not contain; 3 - does not equal; 4 - exists; 5 - does not exist.	tags[0][tag]=target&tags[0][ope
show_tags	Opcja filtra *Show tags*. Possible values: 0 - none; 1 - one; 2 - two; 3 - three.	show_tags=3
tag_name_format	Opcja filtra *Tag name*. Possible values: 0 - full name; 1 - shortened name; 2 - none.	tag_name_format=1
tag_priority	Opcja filtra *Tag display priority*: ciąg znaków rozdzielony przecinkami określający priorytet wyświetlania tagów	tag_priority=customer, target
show_suppressed	Opcja filtra *Show suppressed problems*. Possible values: 1 - show; 2 - do not show.	show_suppressed=1
acknowledgement_status	Opcja filtra *Acknowledgement status*. Possible values: 0 - all; 1 - unacknowledged; 2 - acknowledged.	acknowledgement_status=0
acknowledged_by_me	Opcja filtra *By me*; obsługiwane tylko z `acknowledgement_status=2`. Possible values: 0 - disabled; 1 - enabled.	acknowledged_by_me=1
compact_view	Opcja filtra *Compact view*. Possible values: 0 - disabled; 1 - enabled.	compact_view=1

Parameter	Description	Example
highlight_row	Opcja filtra *Highlight whole row*. Possible values: 0 - disabled; 1 - enabled.	highlight_row=1
filter_name	Opcja właściwości filtra *Name*: dowolny ciąg znaków	filter_name=Databases
filter_show_counter	Opcja właściwości filtra *Show number of records*. Possible values: 0 - disabled; 1 - enabled.	filter_show_counter=1
filter_custom_time	Opcja właściwości filtra *Set custom time period*. Possible values: 0 - disabled; 1 - enabled.	filter_custom_time=1
sort	Kolumna, według której należy sortować. Possible values: clock - sort by *Time* column; host - sort by *Host* column; severity - sort by *Severity* column; name - sort by *Problem* column.	sort=clock
sortorder	Kolejność sortowania wyników. Possible values: DESC - descending; ASC - ascending.	sortorder=DESC
age_state	Opcja filtra *Age less than*; obsługiwane tylko z `show`=3. Possible values: 0 - disable age parameter; 1 - enable age parameter.	age_state=1
age	Opcja filtra *Age less than*: liczba całkowita, liczba dni; obsługiwane tylko z `age_state`=1 i `show`=3.	age=7
groupids	Opcja filtra *Host groups*: tablica identyfikatorów grup hostów.	groupids[]=4
hostids	Opcja filtra *Hosts*: tablica identyfikatorów hostów	hostids[]=10084
triggerids	Opcja filtra *Triggers*: tablica identyfikatorów wyzwalaczy	triggerids[]=22382
show_timeline	Opcja filtra *Show timeline*; nieobsługiwane z `compact_view`=1. Possible values: 0 - do not show; 1 - show.	show_timeline=1
details	Opcja filtra *Show details*. Possible values: 0 - do not show; 1 - show.	details=1
from	Początek zakresu dat, może być względny (np. now-1m); obsługiwane tylko z `filter_custom_time`=1	from=now-2h
to	Koniec zakresu dat, może być względny (np. now-1m); obsługiwane tylko z `filter_custom_time`=1	to=now

Zobacz także: opcje filtra na stronie Problemy.

Tryb kiosku

Tryb kiosku na obsługiwanych stronach frontend można aktywować za pomocą parametrów URL. Na przykład w pulpitach nawigacyjnych:

- `/zabbix.php?action=dashboard.view&kiosk=1` - aktywuje tryb kiosku
- `/zabbix.php?action=dashboard.view&kiosk=0` - aktywuje tryb normalny

Pokaz slajdów

Możliwe jest aktywowanie pokazu slajdów w dashboardzie:

- `/zabbix.php?action=dashboard.view&slideshow=on` - aktywuj pokaz slajdów

Definicje

Przegląd

Chociaż wiele rzeczy we frontend można skonfigurować przy użyciu samego frontend, niektóre dostosowania są obecnie możliwe tylko przez edycję pliku definicji.

Ten plik to `defines.inc.php`, znajdujący się w katalogu /include katalogu dokumentów HTML Zabbix.

Parametry

Parametry w tym pliku, które mogą być interesujące dla użytkowników:

- ZBX_MIN_PERIOD

Minimalny okres wykresu, w sekundach. Domyślnie jedna minuta.

- GRAPH_YAXIS_SIDE_DEFAULT

Domyślne położenie osi Y na prostych wykresach oraz domyślna wartość w polu listy rozwijanej podczas dodawania pozycji do wykresów niestandardowych. Możliwe wartości: 0 - lewa, 1 - prawa.

Domyślnie: 0

- ZBX_SESSION_NAME

Łańcuch używany jako nazwa pliku cookie sesji frontend Zabbixa.

Domyślnie: zbx_sessionid

- ZBX_DATA_CACHE_TTL

Limit czasu TTL w sekundach używany do unieważniania pamięci podręcznej danych odpowiedzi Vault. Ustaw 0, aby wyłączyć buforowanie odpowiedzi Vault.

Domyślnie: 60

- SUBFILTER_VALUES_PER_GROUP

Liczba wartości podfiltra na grupę (na przykład w podfiltrze najnowszych danych).

Domyślnie: 1000

- ZBX_MAX_WIDGET_LINES

Maksymalna liczba wierszy widżetu do wyświetlenia.

Domyślnie: 1000

Dodatkowe języki frontend

Przegląd Aby używać w interfejsie webowym Zabbixa innego języka niż angielski, odpowiednia lokalizacja musi być zainstalowana na serwerze webowym. Dodatkowo do działania tłumaczeń wymagana jest rozszerzenie PHP gettext.

Instalowanie lokalizacji Aby wyświetlić listę wszystkich zainstalowanych języków, uruchom:

```
locale -a
```

Jeśli nie ma na liście niektórych potrzebnych języków, otwórz plik `/etc/locale.gen` i odkomentuj wymagane lokalizacje. Ponieważ Zabbix używa kodowania UTF-8, należy wybrać lokalizacje z zestawem znaków UTF-8.

Teraz uruchom:

```
locale-gen
```

Uruchom ponownie serwer WWW. Należy pamiętać, że jeśli PHP-FPM jest zainstalowany i używany przez serwer WWW, trzeba go również uruchomić ponownie razem z serwerem WWW.

Lokalizacje powinny być teraz zainstalowane. Może być konieczne ponowne wczytanie strony frontend Zabbixa w przeglądarce za pomocą `Ctrl + F5`, aby pojawiły się nowe języki.

Instalowanie Zabbixa Jeśli instalujesz Zabbixa bezpośrednio z repozytorium git Zabbixa, pliki tłumaczeń należy wygenerować ręcznie. Aby wygenerować pliki tłumaczeń, uruchom:

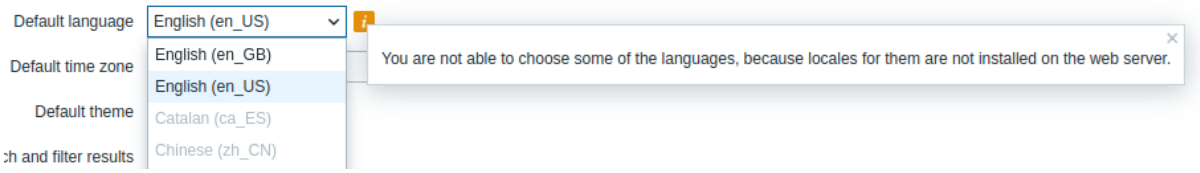
```
make gettext
locale/make_mo.sh
```

Ten krok nie jest potrzebny podczas instalowania Zabbixa z pakietów lub plików źródłowych `tar.gz`.

Wybór języka Istnieje kilka sposobów wyboru języka w interfejsie webowym Zabbix:

- Podczas instalacji interfejsu webowego - w frontend kreatorze instalacji. Wybrany język zostanie ustawiony jako domyślny systemowy.
- Po instalacji domyślny język systemowy można zmienić w *Administracja* > *Ogólne* > *GUI*.
- Język dla konkretnego użytkownika można zmienić w profilu użytkownika.

Jeśli na maszynie nie jest zainstalowany locale dla danego języka, język ten będzie wyszarzony w selektorze języka Zabbix. Jeśli brakuje co najmniej jednego locale, obok selektora języka wyświetlana jest pomarańczowa ikona. Po kliknięciu tej ikony zostanie wyświetlony następujący komunikat: "Nie możesz wybrać niektórych języków, ponieważ nie są dla nich zainstalowane locale na serwerze webowym."



Tworzenie własnego motywu

Przegląd

Domyślnie Zabbix udostępnia kilka predefiniowanych motywów frontend. Aby utworzyć własny motyw, wykonaj przedstawioną tutaj procedurę krok po kroku.

Krok 1: Utwórz swój motyw

Aby utworzyć własny motyw:

1. Utwórz plik CSS od podstaw lub skopiuj istniejący motyw (na przykład `blue-theme.css`, `dark-theme.css` lub inny) z katalogu `assets/styles/` i zmodyfikuj go.
2. Zapisz swój motyw (na przykład `custom-theme.css`) w katalogu `assets/styles/`.

Krok 2: Włączanie motywu na liście motywów

Aby udostępnić niestandardowy motyw we frontendzie, należy dodać go do listy motywów zwracanej przez metodę `APP::getThemes()`. W tym celu należy przesłonić metodę `ZBase::getThemes()` wewnątrz klasy `APP` w następujący sposób:

1. Otwórz plik `include/classes/core/APP.php` i znajdź klasę `APP`:

```
class APP extends ZBase {

}
```

2. Wewnątrz klasy, przed jej końcowym nawiasem zamykającym `}`, dodaj następującą metodę:

```
public static function getThemes() {
        return array_merge(parent::getThemes(), [
            'custom-theme' => _('Custom theme')
        ]);
}
```

gdzie:

- `custom-theme` - wewnętrzny identyfikator motywu, który musi odpowiadać nazwie pliku CSS bez rozszerzenia `.css`;

- `Custom theme` - nazwa wyświetlana, którą zobaczysz we frontendzie.

Aby dodać wiele motywów, wymień je w tablicy, oddzielając poszczególne wpisy przecinkiem. Przecinek po ostatnim wpisie jest opcjonalny.

Na przykład:

```
      public static function getThemes() {
          return array_merge(parent::getThemes(), [
              'custom-theme' => _('Custom theme'),
              'anothertheme' => _('Another theme'),
              'onemoretheme' => _('One more theme')
          ]);
      }
```

Krok 3: Zmiana kolorów wykresu

Aby zmienić kolory wykresu, należy dodać rekord nowego motywu do tabeli bazy danych `graph_theme`.

Przykład dodawania rekordu do bazy danych 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;
```

Krok 4: Używanie nowego motywu

W frontendzie Zabbix możesz ustawić ten motyw jako motyw domyślny lub wybrać go w profilu użytkownika.

Tryb debugowania

Przegląd

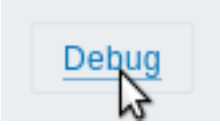
Tryb debugowania może być używany do diagnozowania problemów z wydajnością stron frontend.

Konfiguracja

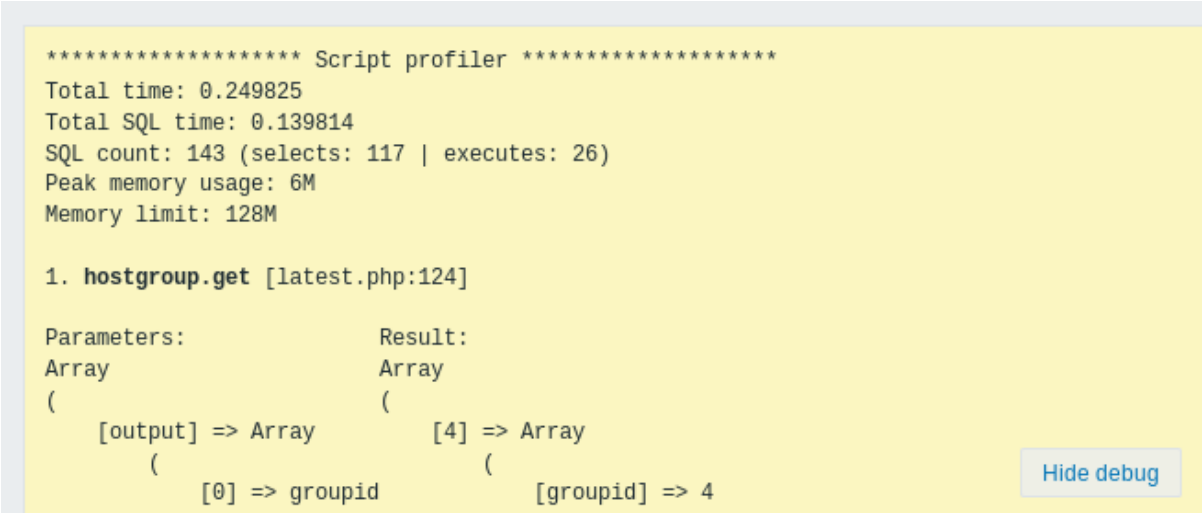
Tryb debugowania można włączyć dla poszczególnych użytkowników należących do grupy użytkowników:

- podczas konfigurowania grupy użytkowników;
- podczas przeglądania skonfigurowanych grup użytkowników.

Gdy dla grupy użytkowników włączony jest *Tryb debugowania*, jej użytkownicy zobaczą przycisk *Debug* w prawym dolnym rogu okna przeglądarki:



Kliknięcie przycisku *Debug* otwiera nowe okno pod zawartością strony, które zawiera statystyki SQL dla strony, a także listę wywołań API i poszczególnych instrukcji SQL:



W przypadku problemów z wydajnością strony to okno może posłużyć do wyszukania przyczyny źródłowej problemu.

Warning:
Włączony *Tryb debugowania* negatywnie wpływa na wydajność frontend.

Pliki cookie używane przez Zabbix

Przegląd

Ta strona zawiera listę plików cookie używanych przez Zabbix.

Nazwa	Opis	Wartości	Wygasa/Max-Age	HttpOnly[1]	Secure[2]
ZBX_SESSION_NAME	Dane sesji frontend Zabbix, przechowywane jako JSON zakodowany przy użyciu base64		Sesja (wygasa po zakończeniu sesji przeglądania)	+	+ (tylko jeśli HTTPS jest włączony na serwerze WWW)

Nazwa	Opis	Wartości	Wygasa/Max-Age	HttpOnly[1]	Secure[2]
tab	Numer aktywnej karty; ten plik cookie jest używany tylko na stronach z wieloma kartami (np. na stronie konfiguracji *Host*, *Trigger* lub *Action*) i jest tworzony, gdy użytkownik przechodzi z karty głównej do innej karty (takiej jak karta *Tags* lub *Dependencies*). 0 jest używane dla karty głównej.	Przykład: 1	Sesja (wygasa po zakończeniu sesji przeglądania)	-	-
browserwarning_ignore	Określa, czy ostrzeżenie o używaniu nieaktualnej przeglądarki powinno zostać zignorowane.	yes	Sesja (wygasa po zakończeniu sesji przeglądania)	-	-
system-message-ok	Komunikat do wyświetlenia natychmiast po ponownym załadowaniu strony.	Komunikat w postaci zwykłego tekstu	Sesja (wygasa po zakończeniu sesji przeglądania) lub natychmiast po ponownym załadowaniu strony	+	-
system-message-error	Komunikat o błędzie do wyświetlenia natychmiast po ponownym załadowaniu strony.	Komunikat w postaci zwykłego tekstu	Sesja (wygasa po zakończeniu sesji przeglądania) lub natychmiast po ponownym załadowaniu strony	+	-

Note:
Wymuszanie flagi 'HttpOnly' dla plików cookie Zabbix za pomocą dyrektywy serwera WWW nie jest obsługiwane.

Przypisy

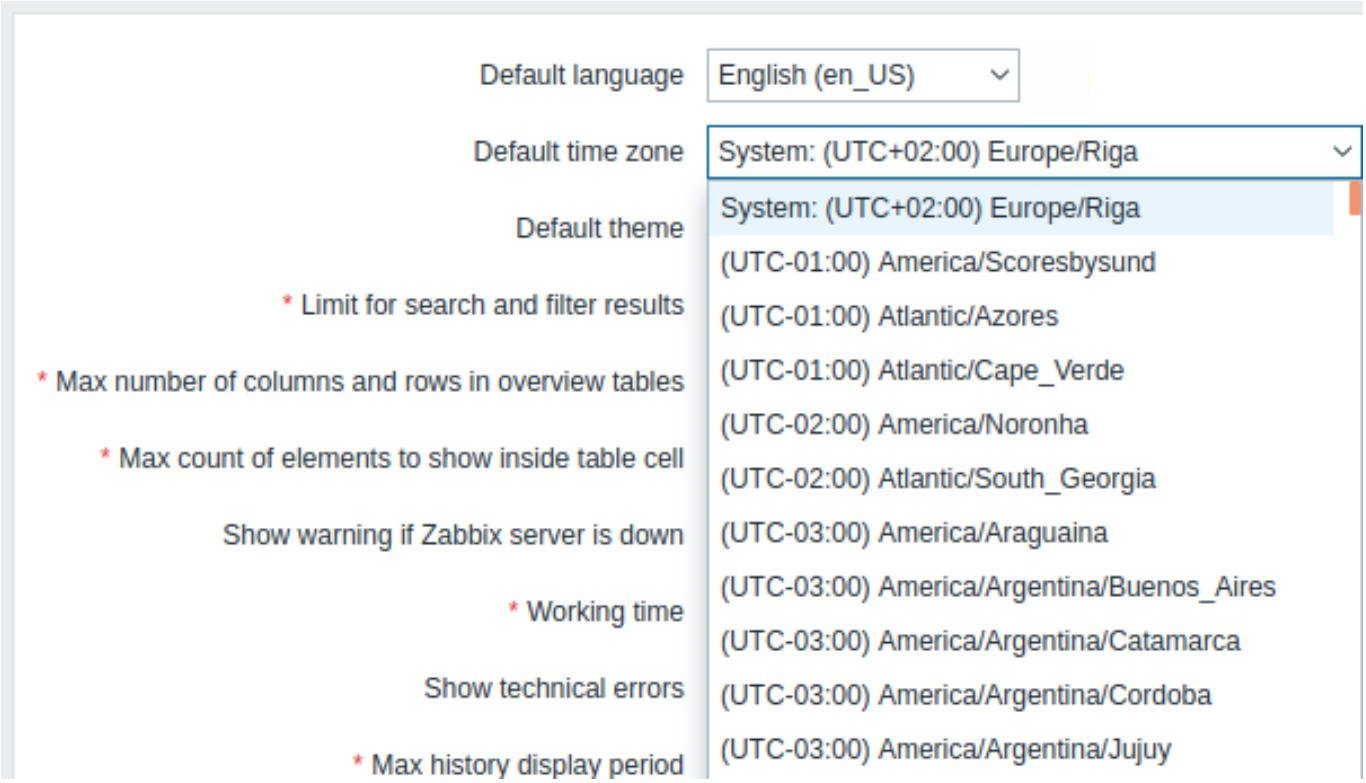
[1] Gdy `HttpOnly` ma wartość 'true', plik cookie będzie dostępny tylko przez protokół HTTP. Oznacza to, że plik cookie nie będzie dostępny dla języków skryptowych, takich jak JavaScript. To ustawienie może skutecznie pomóc ograniczyć kradzież tożsamości poprzez ataki XSS (chociaż nie jest obsługiwane przez wszystkie przeglądarki).

[2] `Secure` wskazuje, że plik cookie powinien być przesyłany wyłącznie przez bezpieczne połączenie HTTPS od klienta. Gdy ma wartość 'true', plik cookie zostanie ustawiony tylko wtedy, gdy istnieje bezpieczne połączenie.

Strefy czasowe

Omówienie

Strefę czasową frontend można ustawić globalnie w frontend i dostosować dla poszczególnych użytkowników.



Jeśli wybrano opcję *System*, dla frontend zostanie użyta strefa czasowa serwera WWW (w tym wartość `date.timezone` w pliku `php.ini`, jeśli została ustawiona), natomiast serwer Zabbix użyje strefy czasowej komputera, na którym działa.

Note:
Serwer Zabbix użyje określonej globalnej strefy czasowej lub strefy czasowej użytkownika wyłącznie podczas rozwijania makr w powiadomieniach (np. `{EVENT.TIME}` może zostać rozwinięte do innej strefy czasowej dla każdego użytkownika) oraz do określania limitu czasu wysyłania powiadomień (zobacz ustawienie „Kiedy aktywne” w konfiguracji mediów użytkownika).

Wybór strefy czasowej nie wpływa na format godziny i daty w frontend. Zamiast tego możesz dostosować język interfejsu (podczas instalacji lub w ustawieniach użytkownika) - wybranie opcji *English (en_US)* włączy również amerykański format godziny i daty w frontend.

Konfiguracja

Globalna strefa czasowa:

- Może zostać ustawiona ręcznie podczas instalacji frontend.
- Może zostać zmodyfikowana w *Administration* > *General* > *GUI*.

Strefa czasowa na poziomie użytkownika:

- Może zostać ustawiona podczas konfigurowania/aktualizowania użytkownika.
- Może zostać ustawiona przez każdego użytkownika w jego profilu użytkownika.

Zobacz także: Dopasowywanie stref czasowych podczas korzystania z interwałów harmonogramu.

Resetowanie hasła

Przegląd Ta sekcja opisuje kroki resetowania haseł użytkowników w Zabbix.

Kroki Jeśli nie pamiętasz hasła do Zabbix i nie możesz się zalogować, skontaktuj się z administratorem Zabbix.

Użytkownik z uprawnieniami administratora Super administrator może zmieniać hasła wszystkich użytkowników w formularzu konfiguracji użytkownika.

Jeśli administrator Super administrator zapomniał hasła i nie może się zalogować, należy uruchomić następujące zapytanie SQL, aby zastosować domyślne hasło do użytkownika Super administrator (zamień „Admin” na odpowiednią nazwę użytkownika Super administrator):

```
UPDATE users SET passwd = '$2a$10$ZXIvHAEP2ZM.dLXTm6uPHOMVlARXX7cqjbhM6Fn0cANzkCQBWpMrS' WHERE username =
```

Po uruchomieniu tego zapytania hasło użytkownika zostanie ustawione na *zabbix*. Pamiętaj, aby zmienić domyślne hasło przy pierwszym logowaniu.

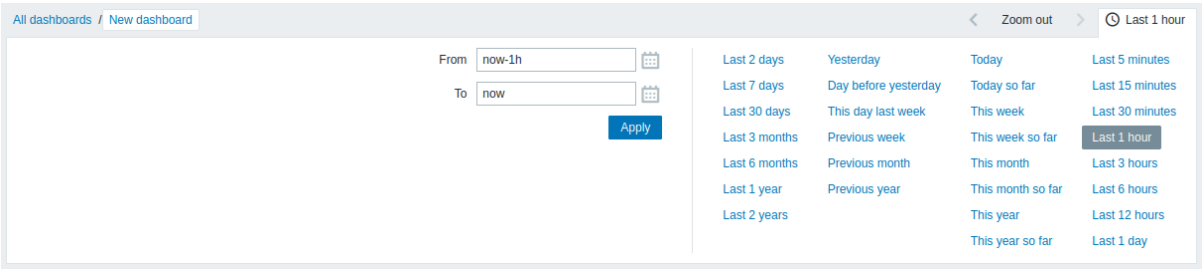
Okres czasu i selektory hostów

Przegląd

Ta strona opisuje dostępne na dashboardach selektory okresu i hosta. Selektory te umożliwiają dynamiczne dostosowanie wyświetlanych danych we wszystkich zgodnych widżetach dashboardu bez edytowania ustawień poszczególnych widżetów.

Selektor okresu

Selektor okresu umożliwia wybranie często wymaganych okresów jednym kliknięciem myszy. Można go rozwinąć lub zwinąć, klikając przycisk selektora okresu w prawym górnym rogu dashboardu.



Opcje takie jak *Dzisiaj*, *Ten tydzień* itd. wyświetlają cały okres, włącznie z godzinami/dniami w przyszłości. Opcje takie jak *Dzisiaj do tej pory*, *Ten tydzień do tej pory* itd. wyświetlają tylko te godziny, które już upłynęły.

Po wybraniu okresu można przesuwać go w czasie do przodu i do tyłu, klikając przyciski strzałek  . Przycisk *Pomniejsz* umożliwia pomniejszenie okresu o 50% w każdym kierunku.

Note:
W przypadku wykresów (z wyjątkiem wykresów z niestandardowym okresem) wyświetlany okres można również wybrać, zaznaczając obszar na wykresie lewym przyciskiem myszy. Po zwolnieniu lewego przycisku myszy wykres zostanie powiększony do zaznaczonego obszaru. Pomniejszenie jest również możliwe przez dwukrotne kliknięcie wykresu.

Pola *Od/Do* wyświetlają wybrany okres w składni czasu bezwzględnego (w formacie `Y-m-d H:i:s`) lub składni czasu względnego. Względny okres może zawierać jedno lub kilka działań matematycznych (- lub +), na przykład `now-1d` lub `now-1d-2h+5m`.

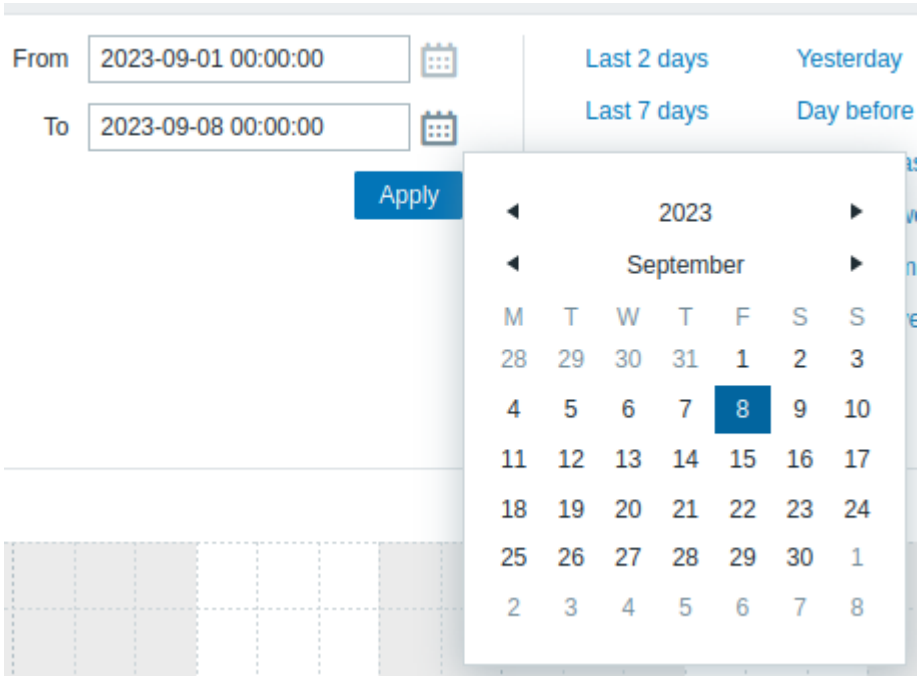
Obsługiwane są następujące skróty czasu względnego:

- `now`
- `s` (sekundy)
- `m` (minuty)
- `h` (godziny)
- `d` (dni)
- `w` (tygodnie)
- `M` (miesiące)
- `y` (lata)

Selektor okresu obsługuje precyzję (na przykład `/M` w `now-1d/M`). Szczegóły dotyczące precyzji:

Precyzja	Od	Do
`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`	Poniedziałek danego tygodnia 00:00:00	Niedziela danego tygodnia 23:59:59
`M`	Pierwszy dzień miesiąca 00:00:00	Ostatni dzień miesiąca 23:59:59
`y`	1 stycznia danego roku 00:00:00	31 grudnia danego roku 23:59:59

Okres można również wybrać za pomocą selektora daty. Aby go otworzyć, kliknij ikonę kalendarza obok pól *Od/Do*.



Note:
W selektorze daty można przechodzić między rokiem, miesiącem i datą za pomocą klawiszy `Tab`, `Shift+Tab` oraz klawiszy strzałek na klawiaturze. Naciśnięcie klawisza `Enter` zatwierdza wybór.

Przykłady:

Od	Do	Wybrany okres
`now/d`	`now/d`	00:00 - 23:59 dzisiaj
`now/d`	`now/d+1d`	00:00 dzisiaj - 23:59 jutro
`now/w`	`now/w`	Poniedziałek 00:00:00 - Niedziela 23:59:59 w tym tygodniu
`now-1y/w`	`now-1y/w`	Tydzień od poniedziałku 00:00:00 do niedzieli 23:59:59 rok temu

Attention:
Użycie ”now/M+1M” dla parametru *Do* może dodać 31 dni, co może skutkować przesunięciem daty o 1-3 dni w zależności od liczby dni w miesiącu. Na przykład, jeśli zostanie użyte w styczniu, wynikiem może być 02 marca zamiast oczekiwanego 28 lutego. Aby uniknąć tego problemu, użyj ”now/M-3d+1M/M”, które prawidłowo uwzględnia długość miesiąca. Podobnie, jeśli konfigurujesz parametr *Od* wstecz, użyj ”now/M+3d-1M/M”.

Selektor hosta

Selektor hosta umożliwia wybranie hosta, którego dane zostaną wyświetlone w zgodnych widżetach pulpitu nawigacyjnego.

Staje się dostępny po edytowaniu i zapisaniu pulpitu nawigacyjnego, jeśli co najmniej jeden widżet jest skonfigurowany z parametrem *Override host* ustawionym tak, aby używać pulpitu nawigacyjnego jako źródła danych.



Parametry konfiguracji: frontend Zabbix

Przegląd Parametry konfiguracji umożliwiają dostosowanie działania frontendu Zabbix.

Określ wartości parametrów w pliku konfiguracji frontendu (`zabbix.conf.php`).

Ta strona ma następującą strukturę:

- Przegląd parametrów (aby wyświetlić pełne informacje o parametrze, kliknij jego nazwę)
- Szczegóły parametrów

Parametr	Opis
$DB['TYPE']	Typ bazy danych.
$DB['SERVER']	Nazwa hosta bazy danych lub adres IP.
$DB['PORT']	Port bazy danych.
$DB['DATABASE']	Nazwa bazy danych używanej przez frontend Zabbix do łączenia się ze skonfigurowaną bazą danych.
$DB['USER']	Nazwa użytkownika używana przez frontend Zabbix do łączenia się z bazą danych.
$DB['PASSWORD']	Hasło używane przez frontend Zabbix do łączenia się z bazą danych.
$DB['SCHEMA']	Nazwa schematu bazy danych używanego w PostgreSQL.
$DB['ENCRYPTION']	Włącza lub wyłącza szyfrowanie TLS połączeń między frontendem Zabbix a bazą danych.
$DB['KEY_FILE']	Ścieżka do pliku prywatnego klucza klienta używanego do połączenia TLS z bazą danych.
$DB['CERT_FILE']	Ścieżka do pliku certyfikatu klienta używanego do połączenia TLS z bazą danych.
$DB['CA_FILE']	Ścieżka do pliku certyfikatu CA używanego do połączenia TLS z bazą danych.
$DB['VERIFY_HOST']	Włącza lub wyłącza weryfikację nazwy hosta serwera bazy danych w certyfikacie TLS.
$DB['CIPHER_LIST']	Lista dozwolonych szyfrów TLS dla połączenia z bazą danych.
$DB['VAULT']	Określa menedżera sekretów Vault używanego do przechowywania danych uwierzytelniających bazy danych.
$DB['VAULT_URL']	Adres URL menedżera sekretów Vault używanego do pobierania danych uwierzytelniających bazy danych.
$DB['VAULT_PREFIX']	Prefiks ścieżki lub zapytania Vault używanego do pobierania danych uwierzytelniających bazy danych.
$DB['VAULT_DB_PATH']	Ścieżka do danych uwierzytelniających bazy danych w menedżerze sekretów Vault.
$DB['VAULT_TOKEN']	Token uwierzytelniający używany do uzyskiwania dostępu do menedżera sekretów Vault.
$DB['VAULT_APP_ROLE_ID']	Identyfikator roli AppRole używany do uwierzytelniania w HashiCorp Vault.
$DB['VAULT_APP_SECRET_ID']	Identyfikator sekretu AppRole używany do uwierzytelniania w HashiCorp Vault.
$DB['VAULT_CERT_FILE']	Ścieżka do pliku certyfikatu klienta używanego do uwierzytelniania w menedżerze sekretów Vault.
$DB['VAULT_KEY_FILE']	Ścieżka do pliku prywatnego klucza klienta używanego do uwierzytelniania w menedżerze sekretów Vault.
$DB['VAULT_CACHE']	Włącza lub wyłącza buforowanie danych uwierzytelniających pobranych z menedżera sekretów Vault.
$ZBX_SERVER	Określa wartość zastępującą nazwę hosta lub adres IP Zabbix.
$ZBX_SERVER_PORT	Określa wartość zastępującą port serwera Zabbix.
$ZBX_SERVER_NAME	Nazwa serwera Zabbix.
$IMAGE_FORMAT_DEFAULT	Domyślny format obrazu.
$HISTORY_PROVIDERS[]	Konfiguracja dostawców przechowywania danych historycznych dla Elasticsearch lub ClickHouse.
$SSO['SETTINGS']	Ustawienia konfiguracji logowania SAML.
$SSO['CERT_STORAGE']	Określa miejsce przechowywania certyfikatów SAML.
$SSO['SP_KEY']	Ścieżka do pliku prywatnego klucza dostawcy usług SAML (SP), gdy klucz prywatny jest przechowywany w systemie plików.
$SSO['SP_CERT']	Ścieżka do pliku certyfikatu dostawcy usług SAML (SP), gdy certyfikat jest przechowywany w systemie plików.
$SSO['IDP_CERT']	Ścieżka do pliku certyfikatu dostawcy tożsamości SAML, gdy certyfikat jest przechowywany w systemie plików.
$ZBX_FEATURE_FLAGS['banners_enabled']	Włącza lub wyłącza obsługę banerów.
$ZBX_FEATURE_FLAGS['http_auth_enabled']	Włącza lub wyłącza uwierzytelnianie użytkowników HTTP.
$ZBX_FEATURE_FLAGS['modules_enabled']	Włącza lub wyłącza dostęp do modułów.
$ZBX_FEATURE_FLAGS['media_type_*']	Włącza lub wyłącza obsługę określonych wstępnie skonfigurowanych typów mediów.
$ZBX_SERVER_TLS['ACTIVE']	Określa, czy TLS jest włączony dla połączeń z serwerem Zabbix.
$ZBX_SERVER_TLS['CA_FILE']	Ścieżka do pliku certyfikatu urzędu certyfikacji (CA) używanego do weryfikacji certyfikatu serwera.
$ZBX_SERVER_TLS['KEY_FILE']	Ścieżka do prywatnego klucza klienta odpowiadającego certyfikatowi klienta.
$ZBX_SERVER_TLS['CERT_FILE']	Ścieżka do pliku certyfikatu klienta, gdy wymagane jest wzajemne uwierzytelnianie TLS.
$ZBX_SERVER_TLS['CERTIFICATE_ISSUER']	Wyróżniona nazwa (DN) wystawcy, która ma zostać dopasowana do certyfikatu serwera.
$ZBX_SERVER_TLS['CERTIFICATE_SUBJECT']	Wyróżniona nazwa (DN) podmiotu, która ma zostać dopasowana do certyfikatu serwera.
$TELEMETRY_PROVIDERS[]	Konfiguracja źródła danych telemetrycznych.

Szczegóły parametrów

$DB['TYPE']

Typ bazy danych.

Wartości: MYSQL - MySQL, POSTGRESQL - PostgreSQL.

$DB['SERVER']

Nazwa hosta bazy danych lub adres IP.

Jeśli parametr $DB['SERVER'] nie jest zdefiniowany, używany jest domyślny gniazdo domeny Unix. W przypadku MySQL ustawienie $DB['SERVER'] na localhost również powoduje połączenie za pośrednictwem domyślnego gniazda Unix.
 W przypadku PostgreSQL parametr $DB['SERVER'] może zawierać wiele wpisów host:port oddzielonych przecinkami. Jeśli dla hosta na liście rozdzielanej przecinkami nie zdefiniowano portu, jako port domyślny używana jest wartość zdefiniowana w $DB['PORT']. Jeśli w parametrze $DB['SERVER'] określono ścieżkę gniazda (na przykład /var/run/pgbouncer), używane jest to gniazdo domeny Unix.

Przykłady:

```
$DB['SERVER'] = 'localhost';
$DB['SERVER'] = 127.0.0.1;
```

$DB['PORT']

Port bazy danych.

Jeśli nie zdefiniowano, używana jest wartość 0.

$DB['DATABASE']

Nazwa bazy danych używana przez frontend Zabbix do łączenia się ze skonfigurowaną bazą danych.

Przykład:

```
$DB['DATABASE'] = 'zabbix';
```

$DB['USER']

Nazwa użytkownika używana przez frontend Zabbix do łączenia się z bazą danych.

$DB['PASSWORD']

Hasło używane przez frontend Zabbix do łączenia się z bazą danych.

$DB['SCHEMA']

Nazwa schematu bazy danych używanego przez PostgreSQL.

$DB['ENCRYPTION']

Włącz lub wyłącz szyfrowanie połączeń między frontendem Zabbix a bazą danych.

Wartości: true - włączone, false - wyłączone.

$DB['KEY_FILE']

Ścieżka do pliku klucza prywatnego klienta używanego do połączenia TLS z bazą danych.

Przykład:

```
// Dla MySQL:
$DB['KEY_FILE'] = '/etc/ssl/mysql/client-key.pem';
```

$DB['CERT_FILE']

Ścieżka do pliku certyfikatu klienta używanego do połączenia TLS z bazą danych.

Przykład:

```
// Dla MySQL:
$DB['CERT_FILE'] = '/etc/ssl/mysql/client-cert.pem';
```

$DB['CA_FILE']

Ścieżka do pliku certyfikatu CA używanego do połączenia TLS z bazą danych.

Przykłady:

```
// Dla MySQL:
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
// Dla PostgreSQL:
$DB['CA_FILE'] = '/etc/ssl/pgsql/root.crt';
```

$DB['VERIFY_HOST']

Włącz lub wyłącz weryfikację nazwy hosta serwera bazy danych w certyfikacie TLS.

Wartości: `true` - włączone, `false` - wyłączone.

$DB['CIPHER_LIST']

Lista dozwolonych szyfrów TLS dla połączenia z bazą danych.

Przykłady:

```
// Dla MySQL:
$DB['CIPHER_LIST'] = 'TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_1
```

$DB['VAULT']

Określa menedżera sekretów Vault używanego do przechowywania danych uwierzytelniających bazy danych.

Wartości: `CyberArk` lub `HashiCorp`.

Przykłady:

```
// Dla CyberArk:
$DB['VAULT'] = 'CyberArk';
// Dla HashiCorp:
$DB['VAULT'] = 'HashiCorp';
```

$DB['VAULT_URL']

URL menedżera sekretów Vault używanego do pobierania danych uwierzytelniających do bazy danych.

Przykłady:

```
// Dla CyberArk:
$DB['VAULT_URL'] = 'https://127.0.0.1:1858';
// Dla HashiCorp:
$DB['VAULT_URL'] = 'https://localhost:8200';
```

$DB['VAULT_PREFIX']

Prefiks ścieżki Vault lub zapytania używanego do pobierania danych uwierzytelniających bazy danych.

Przykład:

```
// Dla HashiCorp:
$DB['VAULT_PREFIX'] = '/v1/secret/data/zabbix/';
```

$DB['VAULT_DB_PATH']

Ścieżka do danych uwierzytelniających bazy danych w menedżerze sekretów Vault.

Przykłady:

```
// Dla CyberArk:
$DB['VAULT_DB_PATH'] = 'AppID=foo&Query=Safe=bar;Object=buzz';
// Dla HashiCorp:
$DB['VAULT_DB_PATH'] = 'database';
```

$DB['VAULT_TOKEN']

Token uwierzytelniający używany do uzyskiwania dostępu do menedżera sekretów Vault.

Przykład:

```
// Dla HashiCorp:
$DB['VAULT_TOKEN'] = 'hvs.CAESIIG_PILmULFYOsEyWHxkZ2mF2a8VPKNLE8eHqd4autYGGh4KHGh2cy5aeTY0NFNSaUp3ZnpWbDF1
```

$DB['VAULT_APP_ROLE_ID']

Identyfikator roli VaultAppSecretID używany do uwierzytelniania w HashiCorp Vault.

Przykład:

```
$DB['VAULT_APP_ROLE_ID'] = '6f1567f1-538d-7d70-e8e4-5d3e59471b2e';
```

$DB['VAULT_APP_SECRET_ID']

VaultAppSecretID używany do uwierzytelniania w HashiCorp Vault.

Przykład:

```
$DB['VAULT_APP_SECRET_ID'] = '89dac8b1-f7d6-96a7-2ecf-f94d2de3d6f1';
```

$DB['VAULT_CERT_FILE']

Ścieżka do pliku certyfikatu klienta używanego do uwierzytelniania w menedżerze sekretów Vault.

Przykład:

```
// Dla CyberArk:
$DB['VAULT_CERT_FILE'] = 'conf/certs/cyberark-cert.pem';
```

$DB['VAULT_KEY_FILE']

Ścieżka do pliku klucza prywatnego klienta używanego do uwierzytelniania w menedżerze sekretów Vault.

Przykład:

```
// Dla CyberArk:
$DB['VAULT_KEY_FILE'] = 'conf/certs/cyberark-key.pem';
```

$DB['VAULT_CACHE']

Włącza lub wyłącza buforowanie danych uwierzytelniających pobranych z menedżera sekretów Vault.

Wartości: `true` - włączone, `false` - wyłączone.

$ZBX_SERVER

Określ wartość zastępującą nazwę hosta/adres IP Zabbix.
 Jeśli parametr $ZBX_SERVER zostanie określony bez $ZBX_SERVER_PORT, zostanie użyty port domyślny (10051).

$ZBX_SERVER_PORT

Określ wartość zastępującą port serwera Zabbix.
 Ten parametr jest używany razem z $ZBX_SERVER.

$ZBX_SERVER_NAME

Nazwa serwera Zabbix.

$IMAGE_FORMAT_DEFAULT

Domyślny format obrazu.

Domyślnie: `IMAGE_FORMAT_PNG`
 Wartości: - `IMAGE_FORMAT_PNG` - format PNG - `IMAGE_FORMAT_JPEG` - format JPEG - `IMAGE_FORMAT_GIF` - format GIF

$HISTORY_PROVIDERS[]

Konfiguracja dostawców przechowywania historii dla Elasticsearch lub ClickHouse.

Obsługiwane parametry konfiguracji: - `types` - tablica typów danych przechowywanych w magazynie zewnętrznym; - `provider` - typ dostawcy historii: `elasticsearch` lub `clickhouse`; - `url` - adres URL dostawcy historii; - `db` - nazwa bazy danych (używana przez ClickHouse); - `username` - użytkownik bazy danych (używany przez ClickHouse); - `password` - hasło do bazy danych (używane przez ClickHouse).

Przykłady:

```
// Dla ClickHouse:
$HISTORY_PROVIDERS[] = [
    'types' => ['uint', 'dbl', 'str', 'log', 'text', 'json'],
    'provider' => 'clickhouse',
    'url' => 'http://localhost:8123',
    'db' => 'zabbix',
    'username' => 'zabbix',
    'password' => 'zabbix'
];
// Dla Elasticsearch:
$HISTORY_PROVIDERS[] = [
    'types' => ['uint', 'dbl', 'str', 'log', 'text', 'json'],
    'provider' => 'elasticsearch',
```

```
    'url' => 'http://localhost:9200'
];
```

$SSO['SETTINGS']

Ustawienia konfiguracyjne logowania SAML. Tablica `$SSO['SETTINGS']` musi mieć strukturę oczekiwaną przez bibliotekę *SAML PHP Toolkit* (dostarczaną wraz z Zabbix).

W `$SSO['SETTINGS']` można określić tylko następujące opcje:

- `strict`
- `baseurl`
- `compress`
- `contactPerson`
- `organization`
- `sp` (tylko opcje określone na tej liście)
    - `attributeConsumingService`
    - `x509certNew`
- `idp` (tylko opcje określone na tej liście)
    - `singleLogoutService` (tylko jedna opcja)
        * `responseUrl`
    - `certFingerprint`
    - `certFingerprintAlgorithm`
    - `x509certMulti`
- `security` (tylko opcje określone na tej liście)
    - `signMetadata`
    - `wantNameId`
    - `requestedAuthnContext`
    - `requestedAuthnContextComparison`
    - `wantXMLValidation`
    - `relaxDestinationValidation`
    - `destinationStrictlyMatches`
    - `rejectUnsolicitedResponsesWithInResponseTo`
    - `signatureAlgorithm`
    - `digestAlgorithm`
    - `lowercaseUrlencoding`

Wszystkie pozostałe opcje są pobierane z bazy danych i nie można ich zastąpić. Opcja `debug` zostanie zignorowana.

Jeśli interfejs użytkownika Zabbix znajduje się za proxy lub modułem równoważenia obciążenia, można użyć niestandardowej opcji `use_proxy_headers`:

- `false` (domyślnie) - ignorowanie nagłówków HTTP `X-Forwarded-*`;
- `true` - używanie nagłówków HTTP `X-Forwarded-*` do utworzenia bazowego adresu URL.

Pełny opis dostępnych opcji konfiguracji znajduje się w oficjalnej dokumentacji biblioteki.

Przykład:

```
$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']

Określ, gdzie są przechowywane certyfikaty SAML.

Wartość domyślna: `file`
 Wartości: - `database` - klucz prywatny i certyfikaty są przechowywane w bazie danych; - `file` - klucz prywatny i certyfikaty są przechowywane w systemie plików.

Przykład:

```
$SSO['CERT_STORAGE'] = 'database';
```

$SSO['SP_KEY']

Ścieżka do pliku klucza prywatnego dostawcy usług SAML (SP), gdy klucz prywatny jest przechowywany w systemie plików.

Przykład:

```
$SSO['SP_KEY'] = 'conf/certs/sp.key';
```

$SSO['SP_CERT']

Ścieżka do pliku certyfikatu dostawcy usługi SAML (SP), gdy certyfikat jest przechowywany w systemie plików.

Przykład:

```
$SSO['SP_CERT'] = 'conf/certs/sp.crt';
```

$SSO['IDP_CERT']

Ścieżka do pliku certyfikatu dostawcy tożsamości SAML, gdy certyfikat jest przechowywany w systemie plików.

Przykład:

```
$SSO['IDP_CERT'] = 'conf/certs/idp.crt';
```

$ZBX_FEATURE_FLAGS['banners_enabled']

Włącz lub wyłącz obsługę banerów.

Wartość domyślna: `true`
 Wartości: `true` - włączona, `false` - wyłączona.

$ZBX_FEATURE_FLAGS['http_auth_enabled']

Włącz lub wyłącz uwierzytelnianie użytkowników HTTP.

Wartość domyślna: `true`
 Wartości: `true` - włączone, `false` - wyłączone.

$ZBX_FEATURE_FLAGS['modules_config_enabled']

Włącz lub wyłącz dostęp do modułów.

Wartość domyślna: `true`
 Wartości: `true` - włączone, `false` - wyłączone.

$ZBX_FEATURE_FLAGS['media_type_denylist']

Wyłącz edytowanie określonych wstępnie skonfigurowanych typów mediów.

Wartości: `email`, `script`, `sms`, `webhook`, `push`.

Przykład:

```
$ZBX_FEATURE_FLAGS['media_type_denylist'] = ['webhook','push'];
```

$ZBX_SERVER_TLS['ACTIVE']

Określa, czy połączenie TLS między frontendem Zabbix a serwerem Zabbix jest włączone.

Wartość domyślna: `false`
 Wartości: `true` - włączone, `false` - wyłączone.

$ZBX_SERVER_TLS['CA_FILE']

Ścieżka do pliku certyfikatu urzędu certyfikacji (CA) używanego do weryfikacji certyfikatu serwera.

Przykład:

```
$ZBX_SERVER_TLS['CA_FILE'] = '/etc/zabbix/ssl/ca.crt';
```

$ZBX_SERVER_TLS['KEY_FILE']

Ścieżka do pliku klucza prywatnego klienta odpowiadającego certyfikatowi klienta używanemu do wzajemnego uwierzytelniania TLS.

Przykład:

```
$ZBX_SERVER_TLS['KEY_FILE'] = '/etc/zabbix/ssl/frontend.key';
```

$ZBX_SERVER_TLS['CERT_FILE']

Ścieżka do pliku certyfikatu klienta, gdy wymagane jest wzajemne uwierzytelnianie TLS.

Przykład:

```
$ZBX_SERVER_TLS['CERT_FILE'] = '/etc/zabbix/ssl/frontend.crt';
```

$ZBX_SERVER_TLS['CERTIFICATE_ISSUER']

Nazwa wyróżniająca (DN) wystawcy, która ma zostać porównana z certyfikatem serwera.

Przykład:

```
$ZBX_SERVER_TLS['CERTIFICATE_ISSUER']  = 'CN=MyZabbixCA';
```

$ZBX_SERVER_TLS['CERTIFICATE_SUBJECT']

Distinguished Name (DN) podmiotu, który ma zostać porównany z certyfikatem serwera.

Przykład:

```
$ZBX_SERVER_TLS['CERTIFICATE_SUBJECT'] = 'CN=zabbix-server.example.com';
```

$TELEMETRY_PROVIDERS[]

Konfiguracja źródła danych telemetrycznych.

Obsługiwane parametry konfiguracji dla ClickHouse: - `url` - ustaw adres URL HTTP lub HTTPS używany do łączenia ze źródłem danych telemetrycznych. - `db` - ustaw nazwę bazy danych. - `username` - ustaw nazwę użytkownika używaną do logowania do bazy danych. - `password` - ustaw hasło używane do logowania do bazy danych. - `ssl_cert_file`, `ssl_key_file`, `ssl_key_password`, `ssl_verify_peer`, `ssl_verify_host`, `ssl_ca_location`, `ssl_ca_file` - ustaw te opcje, aby skonfigurować bezpieczne połączenie (TLS/SSL) z bazą danych. - `vault_path` - ustaw ścieżkę do sekretu Vault zawierającego wartości `username` i `password`. Nie używaj razem z opcjami `username` i `password`.

Przykład:

```
$TELEMETRY_PROVIDERS[] = [
    'provider' => 'clickhouse',
    'url' => 'http://localhost:8123',
    'db' => 'zabbix',
    'username' => 'zabbix',
    'password' => 'zabbix'
];
```

API

Omówienie API Zabbix umożliwia programowe pobieranie i modyfikowanie konfiguracji Zabbix oraz zapewnia dostęp do danych historycznych. Jest szeroko używane do:

- Tworzenia nowych aplikacji współpracujących z Zabbix.
- Integracji Zabbix z oprogramowaniem firm trzecich.
- Automatyzacji rutynowych zadań.

API Zabbix jest API opartym na HTTP i jest dostarczane jako część frontend. Wykorzystuje protokół JSON-RPC 2.0, co oznacza dwie rzeczy:

- API składa się z zestawu oddzielnych metod.
- Żądania i odpowiedzi między klientami a API są kodowane przy użyciu formatu JSON.

Więcej informacji o protokole i JSON można znaleźć w specyfikacji JSON-RPC 2.0 oraz na stronie głównej formatu JSON.

Więcej informacji o integrowaniu funkcjonalności Zabbix z aplikacjami Python można znaleźć w bibliotece Python dla Zabbix.

> **Note:**
> Dostęp użytkownika w Zabbix, obejmujący zarówno konfigurację, jak i dane historyczne, zależy od typu użytkownika, przypisanej roli użytkownika oraz grup użytkowników.

Struktura API składa się z szeregu metod, które nominalnie są pogrupowane w oddzielne API. Każda z metod wykonuje jedno konkretne zadanie. Na przykład metoda `host.create` należy do *host* API i służy do tworzenia nowych hostów. Historycznie API bywają czasami określane jako „klasy”.

Większość API zawiera co najmniej cztery metody: `get`, `create`, `update` i `delete`, służące odpowiednio do pobierania, tworzenia, aktualizowania i usuwania danych, ale niektóre API mogą udostępniać zupełnie inny zestaw metod.

Wykonywanie żądań Po skonfigurowaniu frontend możesz używać zdalnych żądań HTTP do wywoływania API. Aby to zrobić, należy wysyłać żądania HTTP POST do pliku `api_jsonrpc.php` znajdującego się w katalogu frontend.
 Na przykład, jeśli frontend Zabbixa jest zainstalowany pod adresem `https://example.com/zabbix`, żądanie HTTP wywołujące metodę `apiinfo.version` może wyglądać następująco:

```
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}'
```

Żądanie musi zawierać nagłówek `Content-Type` ustawiony na jedną z następujących wartości: `application/json-rpc`, `application/json` lub `application/jsonrequest`.

Obiekt żądania musi zawierać następujące właściwości:

- `jsonrpc` - wersja protokołu JSON-RPC używana przez API (Zabbix API implementuje wersję JSON-RPC 2.0);
- `method` - wywoływana metoda API;
- `params` - parametry przekazywane do metody API;
- `id` - dowolny identyfikator żądania (jeśli zostanie pominięty, API traktuje żądanie jako powiadomienie).

Jeśli żądanie jest poprawne, odpowiedź zwrócona przez API powinna wyglądać następująco:

```
{
    "jsonrpc": "2.0",
    "result": "8.0.0",
    "id": 1
}
```

Z kolei obiekt odpowiedzi zawiera następujące właściwości:

- `jsonrpc` - wersja protokołu JSON-RPC;
- `result` - dane zwrócone przez metodę;
- `id` - identyfikator odpowiadającego mu żądania.

Uwierzytelnianie Aby uzyskać dostęp do dowolnych danych w Zabbix, musisz:

- Użyć istniejącego tokenu API utworzonego w frontendzie Zabbix, w sekcji *Users* > *API tokens*, albo utworzonego za pomocą Token API.
- Użyć tokenu uwierzytelniającego uzyskanego metodą `user.login`.

Na przykład, jeśli chcesz uzyskać nowy token uwierzytelniający, logując się jako standardowy użytkownik *Admin*, żądanie JSON wyglądałoby następująco:

```
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}
```

Jeśli dane uwierzytelniające zostały podane poprawnie, odpowiedź zwrócona przez API powinna zawierać token uwierzytelniający użytkownika:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

Metody autoryzacji Za pomocą nagłówka ”Authorization”

Wszystkie żądania API wymagają uwierzytelnienia lub tokenu API. Dane uwierzytelniające można przekazać, używając nagłówka Authorization w żądaniu:

```
curl --request POST \
  --url 'https://example.com/zabbix/api_jsonrpc.php' \
  --header 'Authorization: Bearer 0424bd59b807674191e7d77572075f33'
```

Attention:
Jeśli występują problemy z uwierzytelnianiem, zobacz Przekazywanie nagłówka Authorization.

API Zabbix akceptuje nagłówki bez rozróżniania wielkości liter (np. `authorization`, `Authorization` i `AUTHORIZATION` są traktowane tak samo).

Nagłówek Authorization jest obsługiwany w żądaniach między źródłami (CORS).

Za pomocą pliku cookie Zabbix

Plik cookie *"zbx_session"* jest używany do autoryzacji żądania API z interfejsu Zabbix UI wykonywanego przy użyciu JavaScript (z modułu lub niestandardowego widżetu).

Przykładowy przepływ pracy Poniższa sekcja przeprowadzi Cię przez kilka przykładów użycia bardziej szczegółowo.

Pobieranie hostów Teraz masz prawidłowy token uwierzytelniania użytkownika (reprezentowany jako zmienna w poniższych przykładach), którego można użyć do uzyskania dostępu do danych w Zabbix. Na przykład możesz użyć metody `host.get`, aby pobrać identyfikatory, nazwy hostów i interfejsy wszystkich skonfigurowanych hostów:

Żądanie:

```
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` to plik zawierający zapytanie JSON. Zamiast pliku możesz przekazać zapytanie w argumencie `--data`.

`data.json`

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": [
            "hostid",
            "host"
        ],
        "selectInterfaces": [
            "interfaceid",
            "ip"
        ]
    },
    "id": 2
}
```

Obiekt odpowiedzi będzie zawierał żądane dane o hostach:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "interfaces": [
                {
                    "interfaceid": "1",
                    "ip": "127.0.0.1"
                }
            ]
        }
    ],
    "id": 2
}
```

Note:
Ze względów wydajnościowych zawsze zaleca się podawanie właściwości obiektu, które chcesz pobrać. Dzięki temu unikniesz pobierania wszystkich danych.

Tworzenie nowej pozycji Teraz utwórz nową pozycję na hoście "Zabbix server", używając danych uzyskanych z poprzedniego żądania `host.get`. Można to zrobić za pomocą metody `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_":
```

Pomyślna odpowiedź będzie zawierać ID nowo utworzonej pozycji, które można wykorzystać do odwoływania się do niej w kolejnych żądaniach:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 3
}
```

Note:
Metoda `item.create`, podobnie jak inne *metody tworzenia*, może również □□□□□□□ować tablice obiektów i tworzyć wiele pozycji za pomocą jednego wywołania API.

Tworzenie wielu wyzwalaczy Zatem, jeśli *metody tworzenia* akceptują tablice, możesz dodać wiele wyzwalaczy, na przykład ten:

```
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
```

Pomyślna odpowiedź będzie zawierać identyfikatory nowo utworzonych wyzwalaczy:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 4
}
```

Aktualizacja pozycji Włącz pozycję, ustawiając jej status na 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}'
```

Pomyślna odpowiedź będzie zawierać ID zaktualizowanej pozycji:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 5
}
```

Note:
Metoda `item.update`, podobnie jak inne *metody aktualizacji*, może również przyjmować tablice obiektów i aktualizować wiele pozycji jednym wywołaniem API.

Aktualizacja wielu wyzwalaczy Włącz wiele wyzwalaczy, ustawiając ich status na 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
```

Pomyślna odpowiedź będzie zawierać identyfikatory zaktualizowanych wyzwalaczy:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938",
            "13939"
        ]
    },
    "id": 6
}
```

Obsługa błędów Do tej pory wszystko, czego próbowaliście, działało poprawnie. Ale co się stanie, jeśli spróbujesz wykonać nieprawidłowe wywołanie API? Spróbuj utworzyć kolejny host, wywołując `host.create`, ale pomijając wymagany parametr `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,
```

Odpowiedź będzie wtedy zawierać komunikat o błędzie:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32602,
        "message": "Invalid params.",
        "data": "No groups for host \"Linux server\"."
    },
    "id": 7
}
```

Jeśli wystąpił błąd, zamiast właściwości `result` obiekt odpowiedzi będzie zawierać właściwość `error` z następującymi danymi:

- `code` - kod błędu;
- `message` - krótki opis błędu;
- `data` - bardziej szczegółowy komunikat o błędzie.

Błędy mogą występować w różnych przypadkach, na przykład przy użyciu nieprawidłowych wartości wejściowych, przekroczeniu limitu czasu sesji lub próbie dostępu do nieistniejących obiektów. Twoja aplikacja powinna umieć w sposób elegancki obsługiwać tego typu błędy.

Wersje API Aby uprościć wersjonowanie API, od Zabbix 2.0.4 wersja API odpowiada wersji samego Zabbix. Możesz użyć metody `apiinfo.version`, aby sprawdzić wersję API, z którą pracujesz. Może to być przydatne przy dostosowywaniu aplikacji do korzystania z funkcji zależnych od wersji.

Zabbix gwarantuje zgodność wsteczną funkcji w obrębie głównej wersji. W przypadku wprowadzania zmian niezgodnych wstecznie między głównymi wydaniami, Zabbix zwykle oznacza stare funkcje jako przestarzałe w następnym wydaniu, a usuwa je dopiero w wydaniu po nim. Zdarza się jednak, że Zabbix usuwa funkcje między głównymi wydaniami bez zapewnienia jakiejkolwiek zgodności wstecznej. Ważne jest, aby nigdy nie polegać na żadnych przestarzałych funkcjach i jak najszybciej migrować do nowszych alternatyw.

> **Note:**
> Możesz śledzić wszystkie zmiany wprowadzone w API w dzienniku zmian API.

Dalsza lektura Teraz masz już wystarczającą wiedzę, aby rozpocząć pracę z Zabbix API, jednak nie zatrzymuj się na tym. W celu dalszej lektury zalecamy zapoznanie się z listą dostępnych API.

Odniesienie do metody

Ta sekcja zawiera przegląd funkcji udostępnianych przez Zabbix API i pomoże Ci odnaleźć się w dostępnych klasach i metodach.

Monitorowanie API Zabbix umożliwia dostęp do historii i innych danych zebranych podczas monitorowania.

Pulpity nawigacyjne

Zarządzaj pulpitami i na ich podstawie twórz zaplanowane raporty.

API pulpitu | API szablonu pulpitu | API raportu

Klaster o wysokiej dostępności

Pobierz listę węzłów serwera i ich status.

API klastra wysokiej dostępności

Historia

Pobierz wartości historyczne zebrane przez procesy monitorowania Zabbix do prezentacji lub dalszego przetwarzania.

API historii

Trendy

Pobierz wartości trendów obliczone przez serwer Zabbix do prezentacji lub dalszego przetwarzania.

Interfejs API trendów

Wydarzenia

Pobierz zdarzenia generowane przez wyzwalacze, wykrywanie sieci i inne systemy Zabbix do bardziej elastycznego zarządzania sytuacjami lub integrację z narzędziami firm trzecich.

Interfejs API zdarzeń

Problemy

Pobierz problemy zgodnie z podanymi parametrami.

Problem API

Mapy

Konfiguruj mapy, aby tworzyć szczegółowe, dynamiczne reprezentacje Twojej infrastruktury IT.

API mapy

Zadania

Interakcja z menedżerem zadań serwera Zabbix, tworzenie zadań i pobieranie odpowiedzi.

API zadań

APM API Zabbix umożliwia dostęp do danych telemetrycznych gromadzonych na potrzeby monitorowania wydajności aplikacji (APM).

Ślady

Pobieranie śladów i składających się na nie zakresów.

API śladów APM | API zakresów APM

Metryki

Pobieranie punktów danych metryk.

APM metric API

Dzienniki

Pobierz rekordy dziennika.

APM log API

Usługi API Zabbix umożliwia dostęp do danych zebranych podczas monitorowania usług.

Umowa dotycząca poziomu usług

Zdefiniuj Wytyczne Poziomu Usług (SLO), pobierz szczegółowe Wskaźniki Poziomu Usług (SLI) wydajności usługi.

SLA API

Usługi

Zarządzaj usługami na potrzeby monitorowania poziomu usług i pobieraj szczegółowe informacje o SLA dowolnej usługi.

Service API

Zbieranie danych API Zabbix umożliwia zarządzanie konfiguracją systemu monitorowania.

Hosty i grupy hostów

Zarządzaj grupami hostów, hostami i wszystkim, co jest z nimi związane, w tym interfejsami hostów, makrami hostów i okresami konserwacji.

API hosta | API grupy hostów | API makra użytkownika | API mapy wartości | API konserwacji

Elementy

Zdefiniuj elementy do monitorowania.

API elementu

Wyzwalacze

Skonfiguruj wyzwalacze, aby otrzymywać powiadomienia o problemach w systemie. Zarządzaj zależnościami wyzwalaczy.

Trigger API

Wykresy

Edytuj wykresy lub oddzielne elementy wykresu, aby lepiej prezentować zebrane dane.

API Wykresów | API elementu wykresu

Szablony i grupy szablonów

Zarządzaj szablonami i łącz je z hostami lub innymi szablonami.

API szablonów | API grup szablonów | API map wartości

Wykrywanie niskiego poziomu

Skonfiguruj reguły wykrywania niskiego poziomu, jak również prototypy elementów, wyzwalaczy i wykresów do monitorowania jednostek dynamicznych.

API reguł LLD | API prototypu elementu | API prototypu wyzwalacza | API prototypu wykresu | API prototypu hosta

Complex event processing

Create complex event processing rules.

Complex event processing API

Korelacja zdarzeń

Twórz niestandardowe reguły korelacji zdarzeń.

API korelacji

Wykrywanie sieci

Zarządzaj regułami wykrywania na poziomie sieci, aby automatycznie wyszukiwać i monitorować nowe hosty. Uzyskaj pełny dostęp do informacji o wykrytych usługach i hostah.

API reguły wykrywania | API sprawdzenia wykrywania | API wykrycia hosta | API wykrycia usługi

Eksport i import

Eksportuj i importuj dane konfiguracyjne Zabbix do tworzenia kopii zapasowych konfiguracji, migracji lub aktualizacji konfiguracji na dużą skalę.

API konfiguracjiI

Monitorowanie sieci WWW

Skonfiguruj scenariusze internetowe, aby monitorować swoje aplikacje i usługi internetowe.

API monitoringu WWW

Alerty API Zabbix umożliwia zarządzanie akcjami i alertami systemu monitoringu.

Akcje i alerty

Zdefiniuj akcje i operacje, aby powiadamiać użytkowników o określonych zdarzeniach lub automatycznie wykonywać zdalne polecenia. Uzyskaj dostęp do informacji o generowanych alertach i ich odbiorcach.

API akcji | API alarmów

Typy mediów

Skonfiguruj typy mediów i wiele sposobów, na jakie użytkownicy będą otrzymywać alerty.

API typów mediów

Skrypty

Konfiguruj i wykonuj skrypty, które pomogą Ci w codziennych zadaniach.

API skryptu

Użytkownicy API Zabbix umożliwia zarządzanie użytkownikami systemu monitoringu.

Użytkownicy i grupy użytkowników

Dodaj użytkowników, którzy będą mieli dostęp do Zabbix, przypisz ich do grup użytkowników i nadaj uprawnienia. Utwórz role do szczegółowego zarządzania prawami użytkowników.

API użytkowników | API grup użytkowników | API katalogu użytkowników | API ról użytkowników

Tokeny API

Zarządzaj tokenami autoryzacyjnymi.

API tokena

Uwierzytelnianie

Zmień opcje konfiguracji uwierzytelniania.

API uwierzytelniania

Urządzenia

Zarządzaj urządzeniami mobilnymi używanymi z aplikacją mobilną Zabbix.

API urządzeń

Administracja Za pomocą API Zabbix można zmieniać ustawienia administracyjne systemu monitoringu.

Ogólne

Zmień wybrane globalne opcje konfiguracji.

API autorejestracji | API map ikon | API obrazów | API ustawień | API wyrażeń regularnych | API modułów | API konektorów

Dziennik audytu

Śledź zmiany konfiguracji dokonane przez każdego użytkownika.

API dziennika audytu

Porządkowanie

Skonfiguruj porządkowanie.

Housekeeping API

Proxy i grupy proxy

Zarządzaj proxy używanymi w rozproszonej konfiguracji monitorowania.

Proxy API | API grup proxy

Makra

Zarządzaj makrami.

API makr użytkownika

Informacje o Zabbix Pobierz wersję API Zabbix lub czas ostatniej aktualizacji z serwera Zabbix.

API info API | Server info API

Akcja

Ta klasa jest przeznaczona do pracy z akcjami.

Odwołania do obiektów:

- Akcja
- Operacja akcji
  - Wiadomość operacji akcji
  - Warunek operacji akcji
- Operacja odzyskiwania akcji
- Operacja aktualizacji akcji
- Filtr akcji
  - Warunek filtra akcji

Dostępne metody:

- action.create - tworzenie nowych akcji
- action.delete - usuwanie akcji
- action.get - pobieranie akcji
- action.update - aktualizowanie akcji

Obiekt Akcji

Następujące obiekty są bezpośrednio powiązane z API `action`.

Działanie

Obiekt działania ma następujące właściwości.

Property	Type	Description
actionid	ID	Identyfikator działania. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w przypadku operacji aktualizacji

Property	Type	Description
esc_period	string	Domyślny czas trwania kroku operacji. Musi wynosić co najmniej 60 sekund. Akceptowana jest liczba sekund (np. 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h) lub makro użytkownika. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone podczas aktualizacji stanu usługi”
eventsource	integer	Typ zdarzeń obsługiwanych przez działanie. Listę obsługiwanych typów zdarzeń można znaleźć w opisie właściwości `source` zdarzenia. Zachowanie właściwości: - *stała* - *wymagane* w przypadku operacji tworzenia
name	string	Nazwa działania. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia
status	integer	Określa, czy działanie jest włączone, czy wyłączone. Możliwe wartości: 0 - *(domyślnie)* włączone; 1 - wyłączone.
pause_symptoms	integer	Określa, czy eskalacja ma zostać wstrzymana, jeśli zdarzenie jest zdarzeniem objawowym. Możliwe wartości: 0 - Nie wstrzymuj eskalacji w przypadku problemów objawowych; 1 - *(domyślnie)* Wstrzymuj eskalację w przypadku problemów objawowych. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”
pause_suppressed	integer	Określa, czy eskalacja ma być wstrzymywana podczas okresów konserwacji. Możliwe wartości: 0 - Nie wstrzymuj eskalacji; 1 - *(domyślnie)* Wstrzymuj eskalację. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”
notify_if_canceled	integer	Określa, czy należy wysyłać powiadomienie po anulowaniu eskalacji. Możliwe wartości: 0 - Nie wysyłaj powiadomienia po anulowaniu eskalacji; 1 - *(domyślnie)* Wysyłaj powiadomienie po anulowaniu eskalacji. Zachowanie właściwości: - *obsługiwane*, jeśli `eventsource` jest ustawione na „zdarzenie utworzone przez wyzwalacz”

Operacja działania

Obiekt operacji działania definiuje operację, która zostanie wykonana po wykonaniu działania. Ma on następujące właściwości.

Property	Type	Description
operationtype	integer	Typ operacji. Możliwe wartości: 0 - wysłanie wiadomości; 1 - skrypt globalny; 2 - dodanie hosta; 3 - usunięcie hosta; 4 - dodanie do grupy hostów; 5 - usunięcie z grupy hostów; 6 - powiązanie szablonu; 7 - odłączenie szablonu; 8 - włączenie hosta; 9 - wyłączenie hosta; 10 - ustawienie trybu inwentaryzacji hosta; 13 - dodanie znaczników hosta; 14 - usunięcie znaczników hosta. Możliwe wartości, jeśli właściwość `eventsource` obiektu Action jest ustawiona na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone podczas aktualizacji stanu usługi”: 0 - „wysłanie wiadomości”; 1 - „skrypt globalny”. Możliwe wartości, jeśli właściwość `eventsource` obiektu Action jest ustawiona na „zdarzenie wewnętrzne”: 0 - „wysłanie wiadomości”. Zachowanie właściwości: - *required*
esc_period	string	Czas trwania kroku eskalacji w sekundach. Musi być większy niż 60 sekund. Akceptuje liczbę sekund (np. 60, 300), wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`) lub makro użytkownika. Jeśli ustawiono wartość 0 lub 0s, zostanie użyty domyślny okres eskalacji działania. Domyślnie: 0s. Zachowanie właściwości: - *supported*, jeśli właściwość `eventsource` obiektu Action jest ustawiona na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone podczas aktualizacji stanu usługi”
esc_step_from	integer	Krok, od którego należy rozpocząć eskalację. Domyślnie: 1. Zachowanie właściwości: - *supported*, jeśli właściwość `eventsource` obiektu Action jest ustawiona na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone podczas aktualizacji stanu usługi”
esc_step_to	integer	Krok, na którym należy zakończyć eskalację. Domyślnie: 1. Zachowanie właściwości: - *supported*, jeśli właściwość `eventsource` obiektu Action jest ustawiona na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone podczas aktualizacji stanu usługi”

Property	Type	Description
evaltype	integer	Metoda oceny warunku operacji. Możliwe wartości: 0 - *(domyślnie)* And/Or; 1 - And; 2 - Or.
opcommand	object	Skrypt globalny do wykonania. Skrypt globalny musi mieć zdefiniowaną właściwość `scriptid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „skrypt globalny”
opcommand_grp	array	Grupy hostów, na których należy uruchomić skrypty globalne. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_hst` nie jest ustawione
opcommand_hst	array	Host, na którym należy uruchomić skrypty globalne. Hosty muszą mieć zdefiniowaną właściwość `hostid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_grp` nie jest ustawione
opconditions	array	Warunki operacji używane dla działań wyzwalacza. Obiekt warunku operacji jest szczegółowo opisany poniżej.
opgroup	array	Grupy hostów, do których należy dodać hosty. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „dodanie do grupy hostów” lub „usunięcie z grupy hostów”
opmessage	object	Obiekt zawierający dane wiadomości wysyłanej przez operację. Obiekt wiadomości operacji jest szczegółowo opisany poniżej. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „wysłanie wiadomości”
opmessage_grp	array	Grupy użytkowników, do których należy wysłać wiadomości. Grupy użytkowników muszą mieć zdefiniowaną właściwość `usrgrpid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „wysłanie wiadomości”, a `opmessage_usr` nie jest ustawione
opmessage_usr	array	Użytkownicy, do których należy wysłać wiadomości. Użytkownicy muszą mieć zdefiniowaną właściwość `userid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „wysłanie wiadomości”, a `opmessage_grp` nie jest ustawione

Property	Type	Description
optemplate	array	Szablony, które należy powiązać z hostami. Szablony muszą mieć zdefiniowaną właściwość `templateid`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „powiązanie szablonu” lub „odłączenie szablonu”
opinventory	object	Tryb inwentaryzacji, który należy ustawić dla hosta. Inwentaryzacja musi mieć zdefiniowaną właściwość `inventory_mode`. Zachowanie właściwości: - *required*, jeśli `operationtype` jest ustawione na „ustawienie trybu inwentaryzacji hosta”
optag	array	Znaczniki hosta, które należy dodać lub usunąć. Znaczniki muszą mieć zdefiniowaną właściwość `tag`. Właściwość `value` jest opcjonalna. Zachowanie właściwości: - *supported*, jeśli `operationtype` jest ustawione na „dodanie znaczników hosta” lub „usunięcie znaczników hosta”.

Wiadomość operacji akcji

Obiekt wiadomości operacji zawiera dane o wiadomości, która zostanie wysłana przez operację. Ma następujące właściwości.

Właściwość	Type	Opis
default_msg	integer	Czy użyć domyślnego tekstu i tematu wiadomości akcji. Możliwe wartości: 0 - użyj danych z operacji; 1 - *(domyślnie)* użyj danych z typu nośnika.
mediatypeid	ID	ID typu nośnika, który zostanie użyty do wysłania wiadomości. Zachowanie właściwości: - *obsługiwane*, jeśli `operationtype` obiektu Action operation object, Action recovery operation object lub Action update operation object jest ustawione na „send message”, albo jeśli `operationtype` obiektu Action update operation object jest ustawione na „notify all involved”
message	string	Tekst wiadomości operacji. Zachowanie właściwości: - *obsługiwane*, jeśli `default_msg` jest ustawione na „use the data from the operation”
subject	string	Temat wiadomości operacji. Zachowanie właściwości: - *obsługiwane*, jeśli `default_msg` jest ustawione na „use the data from the operation”

Warunek operacji akcji

Obiekt warunku operacji akcji definiuje warunek, który musi zostać spełniony, aby wykonać bieżącą operację. Ma następujące właściwości.

Property	Type	Description
conditiontype	integer	Typ warunku. Możliwe wartości: 14 - zdarzenie potwierdzone. Zachowanie właściwości: - *wymagane*
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagane*
operator	integer	Operator warunku. Możliwe wartości: 0 - *(domyślnie)* =

Dla każdego typu warunku operacji obsługiwane są następujące operatory i wartości.

Condition	Condition name	Supported operators	Expected value
14	Zdarzenie potwierdzone	=	Czy zdarzenie jest potwierdzone. Możliwe wartości: 0 - niepotwierdzone; 1 - potwierdzone.

Operacja odzyskiwania akcji

Obiekt operacji odzyskiwania akcji definiuje operację, która zostanie wykonana po rozwiązaniu problemu. Operacje odzyskiwania są możliwe **tylko** dla akcji wyzwalacza, wewnętrznych i usług. Ma następujące właściwości.

Właściwość	Typ	Opis
operationtype	integer	Typ operacji. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”: 0 - wyślij wiadomość; 1 - skrypt globalny; 11 - powiadom wszystkich zaangażowanych. Możliwe wartości, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie wewnętrzne”: 0 - wyślij wiadomość; 11 - powiadom wszystkich zaangażowanych. Zachowanie właściwości: - *wymagane*
opcommand	object	Skrypt globalny do wykonania. Skrypt globalny musi mieć zdefiniowaną właściwość `scriptid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „skrypt globalny”

Właściwość	Typ	Opis
opcommand_grp	array	Grupy hostów, na których mają być uruchamiane skrypty globalne. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, a `operationtype` jest ustawione na „skrypt globalny”, i `opcommand_hst` nie jest ustawione
opcommand_hst	array	Host, na którym mają być uruchamiane skrypty globalne. Hosty muszą mieć zdefiniowaną właściwość `hostid`. Zachowanie właściwości: - *wymagane*, jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, a `operationtype` jest ustawione na „skrypt globalny”, i `opcommand_grp` nie jest ustawione
opmessage	object	Obiekt zawierający dane o wiadomości wysyłanej przez operację odzyskiwania. Obiekt wiadomości operacji jest opisany szczegółowo powyżej. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość”
opmessage_grp	array	Grupy użytkowników, do których mają być wysyłane wiadomości. Grupy użytkowników muszą mieć zdefiniowaną właściwość `usrgrpid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość” i `opmessage_usr` nie jest ustawione
opmessage_usr	array	Użytkownicy, do których mają być wysyłane wiadomości. Użytkownicy muszą mieć zdefiniowaną właściwość `userid`. Zachowanie właściwości: - *wymagane*, jeśli `operationtype` jest ustawione na „wyślij wiadomość” i `opmessage_grp` nie jest ustawione

Operacja aktualizacji akcji

Obiekt operacji aktualizacji akcji definiuje operację, która zostanie wykonana po zaktualizowaniu problemu (dodaniu komentarza, potwierdzeniu, zmianie ważności lub ręcznym zamknięciu). Operacje aktualizacji są możliwe **tylko** dla akcji wyzwalaczy i usług. Ma następujące właściwości.

Właściwość	Type	Opis
operationtype	integer	Typ operacji. Możliwe wartości: 0 - wyślij wiadomość; 1 - skrypt globalny; 12 - powiadom wszystkich zaangażowanych. Zachowanie właściwości: - *required*

Właściwość	Type	Opis
opcommand	object	Skrypt globalny do wykonania. Skrypt globalny musi mieć zdefiniowaną właściwość `scriptid`. Zachowanie właściwości: - *required* jeśli `operationtype` jest ustawione na „skrypt globalny”
opcommand_grp	array	Grupy hostów, na których mają być uruchamiane skrypty globalne. Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`. Zachowanie właściwości: - *required* jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_hst` nie jest ustawione
opcommand_hst	array	Host, na którym mają być uruchamiane skrypty globalne. Hosty muszą mieć zdefiniowaną właściwość `hostid`. Zachowanie właściwości: - *required* jeśli `eventsource` obiektu Action object jest ustawione na „zdarzenie utworzone przez wyzwalacz”, `operationtype` jest ustawione na „skrypt globalny”, a `opcommand_grp` nie jest ustawione
opmessage	object	Obiekt zawierający dane o wiadomości wysyłanej przez operację aktualizacji. Obiekt wiadomości operacji jest opisany szczegółowo powyżej.
opmessage_grp	array	Grupy użytkowników, do których mają być wysyłane wiadomości. Grupy użytkowników muszą mieć zdefiniowaną właściwość `usrgrpid`. Zachowanie właściwości: - *required* jeśli `operationtype` jest ustawione na „wyślij wiadomość”, a `opmessage_usr` nie jest ustawione
opmessage_usr	array	Użytkownicy, do których mają być wysyłane wiadomości. Użytkownicy muszą mieć zdefiniowaną właściwość `userid`. Zachowanie właściwości: - *required* jeśli `operationtype` jest ustawione na „wyślij wiadomość”, a `opmessage_grp` nie jest ustawione

Filtr akcji

Obiekt filtra akcji definiuje zestaw warunków, które muszą zostać spełnione, aby wykonać skonfigurowane operacje akcji. Ma następujące właściwości.

Właściwość	Typ	Opis
conditions	array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich występowania w formule. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
evaltype	integer	Metoda obliczania warunku filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do obliczania warunków filtra. Wyrażenie zawiera identyfikatory odnoszące się do określonych warunków filtra przez ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do obliczania warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odnoszące się do określonych warunków filtra przez ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „custom expression”

Warunek filtra akcji

Obiekt warunku filtra akcji definiuje konkretny warunek, który musi zostać sprawdzony przed uruchomieniem operacji akcji.

Property	Type	Description
conditiontype	integer	Typ warunku. Możliwe wartości, jeśli `eventsource` obiektu Action jest ustawione na „event created by a trigger”: 0 - grupa hostów; 1 - host; 2 - wyzwalacz; 3 - nazwa zdarzenia; 4 - ważność wyzwalacza; 6 - okres; 13 - szablon hosta; 16 - problem jest wyciszony; 25 - tag zdarzenia; 26 - wartość tagu zdarzenia. Możliwe wartości, jeśli `eventsource` obiektu Action jest ustawione na „event created by a discovery rule”: 7 - adres IP hosta; 8 - typ wykrytej usługi; 9 - port wykrytej usługi; 10 - status wykrywania; 11 - czas działania lub niedostępności; 12 - odebrana wartość; 18 - reguła wykrywania; 19 - kontrola wykrywania; 20 - proxy; 21 - obiekt wykrywania; 29 - grupa proxy. Możliwe wartości, jeśli `eventsource` obiektu Action jest ustawione na „event created by active agent autoregistration”: 20 - proxy; 22 - nazwa hosta; 24 - metadane hosta; 29 - grupa proxy. Możliwe wartości, jeśli `eventsource` obiektu Action jest ustawione na „internal event”: 0 - grupa hostów; 1 - host; 13 - szablon hosta; 23 - typ zdarzenia; 25 - tag zdarzenia; 26 - wartość tagu zdarzenia. Możliwe wartości, jeśli `eventsource` obiektu Action jest ustawione na „event created on service status update”: 25 - tag zdarzenia; 26 - wartość tagu zdarzenia; 27 - usługa; 28 - nazwa usługi. Zachowanie właściwości: - *required*
value	string	Wartość do porównania. Zachowanie właściwości: - *required*

Property	Type	Description
value2	string	Dodatkowa wartość do porównania. Zachowanie właściwości: - *required* if `eventsource` obiektu Action jest ustawione na „event created by a trigger”, `conditiontype` jest ustawione na dowolną możliwą wartość dla akcji wyzwalacza, a typ warunku (patrz poniżej) to „26” - *required* if `eventsource` obiektu Action jest ustawione na „internal event”, `conditiontype` jest ustawione na dowolną możliwą wartość dla akcji wewnętrznych, a typ warunku (patrz poniżej) to „26” - *required* if `eventsource` obiektu Action jest ustawione na „event created on service status update”, `conditiontype` jest ustawione na dowolną możliwą wartość dla akcji usługi, a typ warunku (patrz poniżej) to „26”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale po ponownym ich pobraniu zostanie wygenerowany od nowa.
operator	integer	Operator warunku. Możliwe wartości: 0 - *(default)* równa się; 1 - nie równa się; 2 - zawiera; 3 - nie zawiera; 4 - należy do; 5 - jest większe lub równe; 6 - jest mniejsze lub równe; 7 - nie należy do; 8 - pasuje; 9 - nie pasuje; 10 - Tak; 11 - Nie.

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod action.get i action.create.

Poniżej przedstawiono operatory i wartości obsługiwane dla każdego typu warunku.

Condition	Condition name	Supported operators	Expected value
0	Grupa hostów	równa się, nie równa się	ID grupy hostów.
1	Host	równa się, nie równa się	ID hosta.
2	Wyzwalacz	równa się, nie równa się	ID wyzwalacza.
3	Nazwa zdarzenia	zawiera, nie zawiera	Nazwa zdarzenia.
4	Ważność wyzwalacza	równa się, nie równa się, jest większe lub równe, jest mniejsze lub równe	Ważność wyzwalacza. Listę obsługiwanych ważności wyzwalacza można znaleźć we właściwości `severity` wyzwalacza.
5	Wartość wyzwalacza	równa się	Wartość wyzwalacza. Listę obsługiwanych wartości wyzwalacza można znaleźć we właściwości `value` wyzwalacza.

Condition	Condition name	Supported operators	Expected value
6	Okres	należy do, nie należy do	Czas wyzwolenia zdarzenia jako okres.
7	Adres IP hosta	równa się, nie równa się	Jeden lub kilka zakresów adresów IP do sprawdzenia, oddzielonych przecinkami. Więcej informacji o obsługiwanych formatach zakresów adresów IP można znaleźć w sekcji konfiguracja wykrywania sieci.
8	Typ wykrytej usługi	równa się, nie równa się	Typ wykrytej usługi. Typ usługi odpowiada typowi kontroli wykrywania użytej do wykrycia usługi. Listę obsługiwanych typów można znaleźć we właściwości `type` kontroli wykrywania.
9	Port wykrytej usługi	równa się, nie równa się	Jeden lub kilka zakresów portów, oddzielonych przecinkami.
10	Status wykrywania	równa się	Status wykrytego obiektu. Możliwe wartości: 0 - host lub usługa działa; 1 - host lub usługa nie działa; 2 - host lub usługa wykryta; 3 - host lub usługa utracona.
11	Czas działania lub niedostępności	jest większe lub równe, jest mniejsze lub równe	Czas określający, jak długo wykryty obiekt pozostaje w bieżącym statusie, w sekundach.
12	Odebrane wartości	równa się, nie równa się, jest większe lub równe, jest mniejsze lub równe, zawiera, nie zawiera	Wartość zwrócona podczas wykonywania kontroli wykrywania Zabbix agent, SNMPv1, SNMPv2 lub SNMPv3.
13	Szablon hosta	równa się, nie równa się	ID połączonego szablonu.
16	Problem jest wyciszony	Tak, Nie	Wartość nie jest wymagana: użycie operatora „Tak” oznacza, że problem musi być wyciszony, a „Nie” - niewyciszony.
18	Reguła wykrywania	równa się, nie równa się	ID reguły wykrywania.
19	Kontrola wykrywania	równa się, nie równa się	ID kontroli wykrywania.
20	Proxy	równa się, nie równa się	ID proxy.
21	Obiekt wykrywania	równa się	Typ obiektu, który wywołał zdarzenie wykrywania. Możliwe wartości: 1 - wykryty host; 2 - wykryta usługa.
22	Nazwa hosta	zawiera, nie zawiera, pasuje, nie pasuje	Nazwa hosta. Wyrażenie regularne jest obsługiwane przez operatory *pasuje* i *nie pasuje* w warunkach automatycznej rejestracji.
23	Typ zdarzenia	równa się	Konkretne zdarzenie wewnętrzne. Możliwe wartości: 0 - pozycja w stanie „not supported”; 1 - pozycja w stanie „normal”; 2 - reguła LLD w stanie „not supported”; 3 - reguła LLD w stanie „normal”; 4 - wyzwalacz w stanie „unknown”; 5 - wyzwalacz w stanie „normal”.

Condition	Condition name	Supported operators	Expected value
24	Metadane hosta	zawiera, nie zawiera, pasuje, nie pasuje	Metadane automatycznie zarejestrowanego hosta. Wyrażenie regularne jest obsługiwane przez operatory *pasuje* i *nie pasuje*.
25	Tag	równa się, nie równa się, zawiera, nie zawiera	Tag zdarzenia.
26	Wartość tagu	równa się, nie równa się, zawiera, nie zawiera	Wartość tagu zdarzenia.
27	Usługa	równa się, nie równa się	ID usługi.
28	Nazwa usługi	równa się, nie równa się	Nazwa usługi.
29	Grupa proxy	równa się, nie równa się	ID grupy proxy.

action.create

Opis

`object action.create(object/array actions)`

Ta metoda umożliwia tworzenie nowych akcji.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Akcje do utworzenia.

Oprócz standardowych właściwości akcji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Obiekt filtra akcji dla akcji.
operations	array	Operacje akcji do utworzenia dla akcji.
recovery_operations	array	Operacje odzyskiwania akcji do utworzenia dla akcji.
update_operations	array	Operacje aktualizacji akcji do utworzenia dla akcji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID utworzonych akcji we właściwości `actionids`. Kolejność zwracanych identyfikatorów ID jest zgodna z kolejnością przekazanych akcji.

Przykłady

Utwórz akcję wyzwalacza

Utwórz akcję wyzwalacza, która rozpocznie się, gdy wyzwalacz (zawierający słowo „memory” w nazwie) z hosta „10084” przejdzie do stanu PROBLEM. Akcja będzie miała 4 skonfigurowane operacje. Pierwsza, wykonywana natychmiast operacja wyśle wiadomość do wszystkich użytkowników z grupy użytkowników „7” za pośrednictwem typu mediów „1”. Jeśli zdarzenie nie zostanie rozwiązane w ciągu 30 minut, druga operacja uruchomi skrypt „5” (skrypt o zakresie „Action operation”) na wszystkich hostach z grupy „2”. Jeśli zdarzenie zostanie rozwiązane, operacja odzyskiwania powiadomi wszystkich użytkowników, którzy otrzymali jakiekolwiek wiadomości dotyczące problemu. Jeśli zdarzenie zostanie zaktualizowane, operacja potwierdzenia/aktualizacji powiadomi (z niestandardowym tematem i wiadomością) wszystkich użytkowników, którzy otrzymali jakiekolwiek wiadomości dotyczące problemu.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "17"
        ]
    },
    "id": 1
}
```

Utwórz akcję wykrywania

Utwórz akcję wykrywania, która połączy szablon „10001” z wykrytymi hostami.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Używanie niestandardowego filtra wyrażeń

Utwórz akcję wyzwalacza, która używa niestandardowego wyrażenia - „A and (B or C)” - do oceniania warunków akcji. Gdy wyzwalacz o ważności wyższej lub równej „Warning” z hosta „10084” lub hosta „10106” przejdzie do stanu PROBLEM, akcja wyśle wiadomość do wszystkich użytkowników z grupy użytkowników „7” za pośrednictwem typu nośnika „1”. Identyfikatory formuły „A”, „B” i „C” zostały wybrane arbitralnie.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "18"
        ]
    },
    "id": 1
}
```

Utwórz regułę autorejestracji agent

Utwórz akcję autorejestracji, która dodaje host do grupy hostów „2”, gdy nazwa hosta zawiera „SRV” lub metadane zawierają „AlmaLinux”.

Żądanie:

```
{
    "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
```

```
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            19
        ]
    },
    "id": 1
}
```

Utwórz regułę autorejestracji agent z tagami hosta

Utwórz akcję autorejestracji, która dodaje host do grupy hostów „2” oraz dodaje dwa tagi hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.create",
    "params": {
        "name": "Register Linux servers with tags",
        "eventsource": "2",
        "operations": [
            {
                "operationtype": "4",
                "opgroup": [
                    {
                        "groupid": "2"
                    }
                ]
            },
            {
                "operationtype": "13",
                "optag": [
                    {
                        "tag": "location",
                        "value": "office"
                    },
                    {
                        "tag": "city",
                        "value": "Riga"
                    }
                ]
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            20
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr akcji
- Operacja akcji
- Skrypt

Źródło

CAction::create() w *ui/include/classes/api/services/CAction.php*.

action.delete

Opis

`object action.delete(array actionIds)`

Ta metoda umożliwia usuwanie wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Użytkownik nie będzie mógł usunąć wyzwalacza, który w warunkach filtrowania zawiera niedostępny proxy lub grupę proxy. Więcej informacji zawiera sekcja konfiguracja grup użytkowników (*Lista dostępu do proxy*).

Parametry

`(array)` identyfikatory ID akcji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID usuniętych akcji we właściwości `actionids`.

Przykłady

Usuń wiele akcji

Usuń dwie akcje.

Zapytanie:

```
{
"jsonrpc": "2.0",
"method": "action.delete",
"params": [
"17",
"18"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"actionids": [
"17",
"18"
]
},
"id": 1
}
```

Źródło

CAction::delete() w *ui/include/classes/api/services/CAction.php*.

action.get

Opis

`integer/array action.get(object parameters)`

Metoda umożliwia pobranie akcji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
actionids	ID/array	Zwraca tylko akcje o podanych ID.
groupids	ID/array	Zwraca tylko akcje, które używają podanych grup hostów w warunkach akcji.
hostids	ID/array	Zwraca tylko akcje, które używają podanych hostów w warunkach akcji.
triggerids	ID/array	Zwraca tylko akcje, które używają podanych wyzwalaczy w warunkach akcji.
mediatypeids	ID/array	Zwraca tylko akcje, które używają podanych typów mediów do wysyłania wiadomości.
usrgrpids	ID/array	Zwraca tylko akcje, które są skonfigurowane do wysyłania wiadomości do podanych grup użytkowników.
userids	ID/array	Zwraca tylko akcje, które są skonfigurowane do wysyłania wiadomości do podanych użytkowników.
scriptids	ID/array	Zwraca tylko akcje, które są skonfigurowane do uruchamiania podanych skryptów.
selectFilter	query	Zwraca właściwość `filter` z filtrem warunków akcji.
selectOperations	query	Zwraca właściwość `operations` z operacjami akcji.
selectRecoveryOperations	query	Zwraca właściwość `recovery_operations` z operacjami odzyskiwania akcji.
selectUpdateOperations	query	Zwraca właściwość `update_operations` z operacjami aktualizacji akcji.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `actionid`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został wykorzystany.

Przykłady

Pobieranie akcji wyzwalaczy

Pobierz wszystkie skonfigurowane akcje wyzwalaczy wraz z warunkami akcji i operacjami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectRecoveryOperations": "extend",
        "selectUpdateOperations": "extend",
        "selectFilter": "extend",
        "filter": {
            "eventsource": 0
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "actionid": "3",
            "name": "Zgłaszanie problemów administratorom Zabbix",
            "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": "Wyzwalacz: {TRIGGER.NAME}\r\nStan wyzwalacza: {TRIGGER.STATUS}\r\nPozi
                        "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": "Zaktualizowano: {TRIGGER.NAME}",
                        "message": "{USER.FULLNAME} zaktualizował problem dnia {EVENT.UPDATE.DATE} o {EVEN
                        "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
}
```


Pobierz akcje wykrywania

Pobierz wszystkie skonfigurowane akcje wykrywania wraz z warunkami i operacjami akcji. Filtr używa typu oceny „and”, więc

właściwość `formula` jest pusta, a `eval_formula` jest generowana automatycznie.

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "action.get",
  "params": {
    "output": "extend",
    "selectOperations": "extend",
    "selectFilter": "extend",
    "filter": {
      "eventsource": 1
    }
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "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
}
```


Zobacz także

- Filtr akcji
- Operacja akcji

Źródło

CAction::get() w *ui/include/classes/api/services/CAction.php*.

action.update

Opis

`object action.update(object/array actions)`

Ta metoda umożliwia aktualizację istniejących akcji.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości akcji do zaktualizowania.

Właściwość `actionid` musi być zdefiniowana dla każdej akcji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości akcji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Obiekt filtra akcji, który zastąpi bieżący filtr.
operations	array	Operacje akcji, które zastąpią istniejące operacje.

Parametr	Typ	Opis
recovery_operations	array	Operacje odzyskiwania akcji, które zastąpią istniejące operacje odzyskiwania. Zachowanie parametru: - *obsługiwane*, jeśli `eventsource` obiektu akcji jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
update_operations	array	Operacje aktualizacji akcji, które zastąpią istniejące operacje aktualizacji. Zachowanie parametru: - *obsługiwane*, jeśli `eventsource` obiektu akcji jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID zaktualizowanych akcji we właściwości `actionids`.

Przykłady

Wyłącz akcję

Wyłącz akcję, to znaczy ustaw jej status na „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "action.update",
    "params": {
        "actionid": "2",
        "status": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "actionids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr akcji
- Operacja akcji

Źródło

CAction::update() in *ui/include/classes/api/services/CAction.php*.

Alarm

Ta klasa jest przeznaczona do pracy z alarmami.

Odwołania do obiektu:

- Alert

Dostępne metody:

- alert.get - pobieranie alarmów

Alert obiektu

Następujące obiekty są bezpośrednio powiązane z API `alert`.

Alert

> **Note:**
> Alerty są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt alertu zawiera informacje o tym, czy określone operacje akcji zostały wykonane pomyślnie. Ma następujące właściwości.

Właściwość	Typ	Opis
alertid	ID	ID alertu.
actionid	ID	ID akcji, która wygenerowała alert.
alerttype	integer	Typ alertu. Możliwe wartości: 0 - wiadomość; 1 - zdalne polecenie.
clock	timestamp	Czas wygenerowania alertu.
error	string	Tekst błędu, jeśli wystąpiły problemy z wysłaniem wiadomości lub uruchomieniem polecenia.
esc_step	integer	Krok eskalacji akcji, podczas którego został wygenerowany alert.
eventid	ID	ID zdarzenia, które wyzwoliło akcję.
mediatypeid	ID	ID typu nośnika, który został użyty do wysłania wiadomości.
message	text	Treść wiadomości. Zachowanie właściwości: - *obsługiwane* jeśli `alerttype` jest ustawione na ”message”
retries	integer	Liczba prób wysłania wiadomości przez Zabbix.
sendto	string	Adres, nazwa użytkownika lub inny identyfikator odbiorcy. Zachowanie właściwości: - *obsługiwane* jeśli `alerttype` jest ustawione na ”message”
status	integer	Status wskazujący, czy operacja akcji została wykonana pomyślnie. Możliwe wartości, jeśli `alerttype` jest ustawione na ”message”: 0 - wiadomość nie została wysłana; 1 - wiadomość została wysłana; 2 - niepowodzenie po określonej liczbie prób; 3 - nowy alert nie został jeszcze przetworzony przez menedżera alertów. Możliwe wartości, jeśli `alerttype` jest ustawione na ”remote command”: 0 - polecenie nie zostało uruchomione; 1 - polecenie zostało uruchomione; 2 - podjęto próbę uruchomienia polecenia na Zabbix agent, ale był niedostępny.
subject	string	Temat wiadomości. Zachowanie właściwości: - *obsługiwane* jeśli `alerttype` jest ustawione na ”message”
userid	ID	ID użytkownika, do którego wysłano wiadomość.
p_eventid	ID	ID zdarzenia problemu, które wygenerowało alert.
acknowledgeid	ID	ID potwierdzenia, które wygenerowało alert.

alert.get

Opis

`integer/array alert.get(object parameters)`

Metoda umożliwia pobieranie alertów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
alertids	ID/array	Zwraca tylko alerty o podanych identyfikatorach.
actionids	ID/array	Zwraca tylko alerty wygenerowane przez podane akcje.
eventids	ID/array	Zwraca tylko alerty wygenerowane przez podane zdarzenia.
groupids	ID/array	Zwraca tylko alerty wygenerowane przez obiekty z podanych grup hostów.
hostids	ID/array	Zwraca tylko alerty wygenerowane przez obiekty z podanych hostów.
mediatypeids	ID/array	Zwraca tylko alerty wiadomości, w których użyto podanych typów mediów.
objectids	ID/array	Zwraca tylko alerty wygenerowane przez podane obiekty
userids	ID/array	Zwraca tylko alerty wiadomości, które zostały wysłane do podanych użytkowników.
eventobject	integer	Zwraca tylko alerty wygenerowane przez zdarzenia powiązane z obiektami podanego typu. Listę obsługiwanych typów obiektów można znaleźć w `object` zdarzenia. Domyślnie: 0 - wyzwalacz.
eventsource	integer	Zwraca tylko alerty wygenerowane przez zdarzenia podanego typu. Listę obsługiwanych typów zdarzeń można znaleźć w `source` zdarzenia. Domyślnie: 0 - zdarzenia wyzwalacza.
time_from	timestamp	Zwraca tylko alerty wygenerowane po podanym czasie.
time_till	timestamp	Zwraca tylko alerty wygenerowane przed podanym czasem.
selectHosts	query	Zwraca właściwość `hosts` z danymi hostów, które wyzwoliły operację akcji.
selectMediatypes	query	Zwraca właściwość `mediatypes` z tablicą typów mediów użytych dla alertu wiadomości. Ograniczenia zależne od typu użytkownika można znaleźć w `mediatype.get`.
selectUsers	query	Zwraca właściwość `users` z tablicą użytkowników, do których była adresowana wiadomość. Ograniczenia zależne od typu użytkownika można znaleźć w `user.get`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `alertid`, `clock`, `eventid`, `mediatypeid`, `sendto`, `status`.
countOutput	boolean	Parametry te są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	

Parametr	Typ	Opis
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów albo;
- liczbę zwracanych obiektów, jeśli parametr `countOutput` został użyty.

Przykłady

Pobieranie alertów według ID akcji

Pobierz wszystkie alerty wygenerowane przez akcję „3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "alert.get",
    "params": {
        "output": "extend",
        "actionids": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "alertid": "1",
            "actionid": "3",
            "eventid": "21243",
            "userid": "1",
            "clock": "1362128008",
            "mediatypeid": "1",
            "sendto": "support@company.com",
            "subject": "PROBLEM: Zabbix agent na serwerze Linux jest niedostępny przez 5 minut: ",
            "message": "Wyzwalacz: Zabbix agent na serwerze Linux jest niedostępny przez 5 minut: \nStan w
            "status": "0",
            "retries": "3",
            "error": "",
            "esc_step": "1",
            "alerttype": "0",
            "p_eventid": "0",
            "acknowledgeid": "0"
        }
    ],
    "id": 1
}
```

Zobacz także

- Host
- Typ nośnika
- Użytkownik

Źródło

CAlert::get() in *ui/include/classes/api/services/CAlert.php*.

API info

Klasa jest przeznaczona do pobierania metainformacji o interfejsie API.

Dostępne metody:

- apiinfo.version - pobieranie wersji API Zabbix

apiinfo.version

Opis

`string apiinfo.version(array)`

Ta metoda umożliwia pobranie wersji Zabbix API.

> **Attention:**
> Ta metoda jest dostępna wyłącznie dla użytkowników nieuwierzytelnionych.

Parametry

`(array)` Metoda akceptuje pustą tablicę.

Zwracane wartości

`(string)` Zwraca wersję API Zabbix.

> **Note:**
> Zaczynając od wersji Zabbix 2.0.4, wersja API jest taka sama jak wersja Zabbix.

Przykłady

Pobieranie wersji API

Pobierz wersję API Zabbix.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "apiinfo.version",
    "params": [],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "8.0.0",
    "id": 1
}
```

Źródło

CAPIInfo::version() in *ui/include/classes/api/services/CAPIInfo.php*.

APM span

Ta klasa jest przeznaczona do pracy ze spanami zbieranymi na potrzeby monitorowania wydajności aplikacji (APM).

Odwołania do obiektów:

- APM span

Dostępne metody:

- apmspan.get - pobieranie spanów

Obiekt span APM

Poniższe obiekty są bezpośrednio powiązane z API `apmspan`.

Note:
Każda właściwość odpowiada kolumnie w tabeli `otel_traces` źródła danych telemetrycznych. Listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych.

Span APM

Obiekt span APM reprezentuje span (pojedynczą operację w ramach śladu, taką jak żądanie HTTP, zapytanie do bazy danych lub wywołanie funkcji).

Spany tworzą hierarchię rozpoczynającą się od spanu głównego, który reprezentuje pierwszą operację żądania. Każdy inny span jest spanem podrzędnym spanu, podczas którego rozpoczęto jego operację.

Obiekt span APM ma następujące właściwości:

Właściwość	Typ	Opis
timestamp	integer	Czas uniksowy (w nanosekundach), w którym rozpoczął się span. Zachowanie właściwości: - *tylko do odczytu*
traceid	string	Identyfikator śladu, do którego należy span. Zachowanie właściwości: - *tylko do odczytu*
spanid	string	Identyfikator spanu. Zachowanie właściwości: - *tylko do odczytu*
parent_spanid	string	Identyfikator spanu nadrzędnego. Pusty dla spanu głównego. Zachowanie właściwości: - *tylko do odczytu*
trace_state	string	Dodatkowe dane, jeśli istnieją, dołączane do śladu przez narzędzia śledzenia (zobacz W3C Trace Context). Zachowanie właściwości: - *tylko do odczytu*
span_name	string	Nazwa operacji reprezentowanej przez span (np. `mysqli_query`). Zachowanie właściwości: - *tylko do odczytu*
span_kind	string	Rola spanu. Możliwe wartości: `Unspecified` - rodzaj spanu nie został określony; `Internal` - operacja wykonana w aplikacji; `Server` - obsługa przychodzącego żądania; `Client` - żądanie wychodzące; `Producer` - utworzenie operacji asynchronicznej (np. wysłanie wiadomości); `Consumer` - przetwarzanie operacji asynchronicznej. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
service_name	string	Nazwa usługi, która utworzyła span (np. `zabbix-frontend`), pobrana z atrybutu zasobu OpenTelemetry `service.name`. Zachowanie właściwości: - *tylko do odczytu*
resource_attributes	object	Atrybuty zasobu spanu w postaci klucz-wartość (np. `host.name: eabf8c2a5ddc`). Zachowanie właściwości: - *tylko do odczytu*
scope_name	string	Nazwa zakresu instrumentacji, który utworzył span. Zachowanie właściwości: - *tylko do odczytu*
scope_version	string	Wersja zakresu instrumentacji. Zachowanie właściwości: - *tylko do odczytu*
span_attributes	object	Atrybuty spanu w postaci klucz-wartość (np. `db.system: mysql`). Zachowanie właściwości: - *tylko do odczytu*
duration	integer	Czas trwania spanu w nanosekundach. Zachowanie właściwości: - *tylko do odczytu*
status_code	string	Stan operacji reprezentowanej przez span. Możliwe wartości: `Unset` - spanowi nie przypisano jawnego stanu; `Ok` - operacja została ukończona pomyślnie; `Error` - operacja została ukończona z błędem. Zachowanie właściwości: - *tylko do odczytu*
status_message	string	Dodatkowe informacje, jeśli istnieją, dotyczące stanu spanu. Zachowanie właściwości: - *tylko do odczytu*
events	array	Tablica obiektów zdarzeń spanu, czyli istotnych momentów podczas operacji (np. wyjątków). Zachowanie właściwości: - *tylko do odczytu*
links	array	Tablica obiektów łączy spanu wskazujących powiązane spany w tym samym lub innym śladzie. Zachowanie właściwości: - *tylko do odczytu*

apmspan.get

Opis

`integer/array apmspan.get(object parameters)`

Metoda umożliwia pobieranie spanów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
time_from	timestamp	Zwraca tylko zakresy, które rozpoczęły się o podanym czasie lub później. Zachowanie parametru: - *required*
time_till	timestamp	Zwraca tylko zakresy, które rozpoczęły się przed podanym czasem. Zachowanie parametru: - *required*
traceids	string/array	Zwraca tylko zakresy należące do podanych śladów.
spanids	string/array	Zwraca tylko zakresy o podanych identyfikatorach.
parent_spanids	string/array	Zwraca tylko zakresy z podanymi zakresami nadrzędnymi.
resource_attributes	object/array	Zwraca tylko zakresy z podanymi atrybutami zasobu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie zakresy. Możliwe wartości operatora: 0 - *(default)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
resource_attributes_evaltype	integer	Metoda oceny warunków w `resource_attributes`. Możliwe wartości: 0 - *(default)* And/Or; 2 - Or.
span_attributes	object/array	Zwraca tylko zakresy z podanymi atrybutami zakresu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie zakresy. Możliwe wartości operatora: 0 - *(default)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
span_attributes_evaltype	integer	Metoda oceny warunków w `span_attributes`. Możliwe wartości: 0 - *(default)* And/Or; 2 - Or.
min_duration	integer	Zwraca tylko zakresy, których czas trwania jest większy lub równy podanej wartości, w nanosekundach.
max_duration	integer	Zwraca tylko zakresy, których czas trwania jest krótszy niż podana wartość, w nanosekundach.

Parameter	Type	Description
filter	object	Zwraca tylko wyniki dokładnie pasujące do podanego filtra. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncze wartości lub tablice wartości, względem których wykonywane jest dopasowanie. Obsługiwane właściwości: `traceid`, `spanid`, `parent_spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `status_code`, `status_message`.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są ciągi znaków, których należy szukać. Jeśli nie podano dodatkowych opcji, wyszukiwanie będzie wykonywane za pomocą `LIKE "%…%"`. Obsługiwane właściwości: `trace_state`, `span_name`, `service_name`, `scope_name`, `scope_version`, `status_message`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `timestamp`, `traceid`, `spanid`, `parent_spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `duration`, `status_code`, `status_message`.
offset	integer	Liczba rekordów pomijanych przed zwróceniem wyniku.
countOutput	boolean	Opis tych parametrów znajduje się w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie hierarchii spanów śladu

Pobierz pierwsze cztery spany śladu „dee625f33cb0994e86c5cd4640e3b09c", posortowane według czasu rozpoczęcia. Właściwość `parent_spanid` pokazuje zależności między spanami: span główny nie ma elementu nadrzędnego, a każdy inny span odwołuje się do spanu, który go uruchomił.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Ślad APM
- Dziennik APM
- Metryka APM

Źródło

CApmSpan::get() w *ui/include/classes/api/services/CApmSpan.php*.

Autorejestracja

Ta klasa jest przeznaczona do pracy z autorejestracją.

Odniesienia do obiektów:

- Autoregistration

Dostępne metody:

- autoregistration.get - pobierz autorejestrację
- autoregistration.update - aktualizuj autorejestrację

Obiekt Autorejestracja

Następujące obiekty są bezpośrednio powiązane z API `autoregistration`.

Autorejestracja

Obiekt autorejestracji ma następujące właściwości.

Właściwość	Type	Opis
tls_accept	integer	Typ dozwolonych połączeń przychodzących dla autorejestracji. Możliwe wartości: 1 - zezwalaj na połączenia nieszyfrowane; 2 - zezwalaj na TLS z PSK; 3 - zezwalaj zarówno na połączenia nieszyfrowane, jak i TLS z PSK.
tls_psk_identity	string	Tożsamość PSK; musi być sparowana tylko z jednym PSK (w obrębie autoregistration, hosts i proxies). Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK należy użyć. Zachowanie właściwości: - *tylko do zapisu*
tls_psk	string	Klucz współdzielony (PSK); musi mieć co najmniej 32 cyfry szesnastkowe. Zachowanie właściwości: - *tylko do zapisu*

autoregistration.get

Opis

`object autoregistration.get(object parameters)`

Metoda umożliwia pobranie obiektu autorejestracji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Typ	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Zwracane wartości

`(object)` Zwraca obiekt rejestracji automatycznej.

Przykład

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "autoregistration.get",
  "params": {
    "output": "extend"
```

```
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tls_accept": "3"
  },
  "id": 1
}
```

Źródło

CAutoregistration::get() w *ui/include/classes/api/services/CAutoregistration.php*.

autoregistration.update

Opis

`object autoregistration.update(object autoregistration)`

Ta metoda umożliwia aktualizację istniejącej autorejestracji.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Właściwości rejestracji automatycznej do aktualizacji.

Zwracane wartości

`(boolean )` Zwraca wartość logiczną true jako wynik po udanej aktualizacji.

Przykłady

Zapytanie:

```
{
    "jsonrpc": "2.0",
    "method": "autoregistration.update",
    "params": {
        "tls_accept": "3",
        "tls_psk_identity": "PSK 001",
        "tls_psk": "11111595725ac58dd977beef14b97461a7c1045b9a1c923453302c5473193478"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Źródło

CAutoregistration::update() w *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
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*

Property	Type	Description
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”
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*

Property	Type	Description
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.
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”

Property	Type	Description
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”
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*

Property	Type	Description
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*
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*.

Connector

Ta klasa jest przeznaczona do pracy z connectorami.

Odwołania do obiektów:

- Connector
- Filtr tagów

Dostępne metody:

- connector.create - tworzenie nowych connectorów
- connector.delete - usuwanie connectorów
- connector.get - pobieranie connectorów
- connector.update - aktualizacja connectorów

Obiekt łącznika

Następujące obiekty są bezpośrednio powiązane z API `connector`.

Connector

Obiekt connector ma następujące właściwości.

Property	Type	Description
connectorid	ID	ID connectora. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* w operacjach aktualizacji
name	string	Nazwa connectora. Zachowanie właściwości: - *wymagana* w operacjach tworzenia
url	string	Adres URL punktu końcowego, czyli adres URL odbiorcy. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *wymagana* w operacjach tworzenia
protocol	integer	Protokół komunikacyjny. Możliwe wartości: 0 - *(domyślnie)* Zabbix Streaming Protocol v1.0.
data_type	integer	Typ danych. Możliwe wartości: 0 - *(domyślnie)* Wartości pozycji; 1 - Zdarzenia.
item_value_type	integer	Suma typów wartości pozycji, które mają być wysyłane. Możliwe wartości: 1 - Liczbowe (zmiennoprzecinkowe); 2 - Znakowe; 4 - Dziennik; 8 - Liczbowe (bez znaku); 16 - Tekstowe; 32 - Binarne; 64 - JSON. Domyślnie: 31 - Wszystkie typy pozycji (z wyjątkiem binarnych i JSON). Zachowanie właściwości: - *obsługiwana*, jeśli `data_type` ma wartość „Item values”.

Property	Type	Description
max_records	integer	Maksymalna liczba zdarzeń lub pozycji, które mogą zostać wysłane w jednej wiadomości. Możliwe wartości: 0-2147483647 (maksymalna wartość 32-bitowej liczby całkowitej ze znakiem). Domyślnie: 0 - Bez ograniczeń.
max_senders	integer	Liczba procesów nadawcy uruchamianych dla tego connectora. Możliwe wartości: 1-100. Domyślnie: 1.
max_attempts	integer	Liczba prób. Możliwe wartości: 1-5. Domyślnie: 1.
attempt_interval	string	Odstęp między kolejnymi próbami. Akceptowane są sekundy. Możliwe wartości: 0s-10s. Domyślnie: 5s. Zachowanie właściwości: - *obsługiwana*, jeśli `max_attempts` jest większe niż 1.
timeout	string	Limit czasu. Obsługiwane są przyrostki czasu (np. `30s`, `1m`). Obsługiwane są makra użytkownika. Możliwe wartości: 1s-60s. Domyślnie: 5s.
http_proxy	string	Ciąg połączenia proxy HTTP(S) podany jako *[protocol]://[username[:password]@]proxy.example.com[:port]*. Obsługiwane są makra użytkownika.
authtype	integer	Metoda uwierzytelniania HTTP. Możliwe wartości: 0 - *(domyślnie)* Brak; 1 - Basic; 2 - NTLM; 3 - Kerberos; 4 - Digest; 5 - Bearer.
username	string	Nazwa użytkownika. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *obsługiwana*, jeśli `authtype` ma wartość „Basic”, „NTLM”, „Kerberos” lub „Digest”
password	string	Hasło. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *obsługiwana*, jeśli `authtype` ma wartość „Basic”, „NTLM”, „Kerberos” lub „Digest”

Property	Type	Description
token	string	Token Bearer. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *wymagana*, jeśli `authtype` ma wartość „Bearer”
verify_peer	integer	Czy sprawdzać, czy certyfikat hosta jest autentyczny. Możliwe wartości: 0 - Nie sprawdzaj; 1 - *(domyślnie)* Sprawdzaj.
verify_host	integer	Czy sprawdzać, czy nazwa hosta używana dla połączenia jest zgodna z nazwą w certyfikacie hosta. Możliwe wartości: 0 - Nie sprawdzaj; 1 - *(domyślnie)* Sprawdzaj.
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Obsługiwane są makra użytkownika.
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Obsługiwane są makra użytkownika.
ssl_key_password	string	Hasło do pliku klucza SSL. Obsługiwane są makra użytkownika.
description	text	Opis connectora.
status	integer	Czy connector jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
tags_evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.

Filtr tagów

Filtr tagów umożliwia eksport tylko pasujących wartości pozycji lub zdarzeń. Jeśli nie zostanie ustawiony, eksportowane będą wszystkie dane. Obiekt filtra tagów ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu. Property behavior: - *required*
operator	integer	Operator warunku. Możliwe wartości: 0 - *(default)* Równe; 1 - Nie równe; 2 - Zawiera; 3 - Nie zawiera; 12 - Istnieje; 13 - Nie istnieje.
value	string	Wartość tagu.

connector.create

Opis

`object connector.create(object/array connectors)`

Ta metoda umożliwia tworzenie nowych obiektów konektora.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Obiekty connectorów do utworzenia.

Oprócz standardowych właściwości connectora metoda akceptuje następujące parametry.

Parametr	Type	Opis
tags	array	Filtr tagów connectora.

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID utworzonych konektorów we właściwości `connectorids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych konektorów.

Przykłady

Tworzenie konektora

Utwórz konektor, aby eksportować zdarzenia wyzwalaczy z filtrem tagów. Uwierzytelnianie HTTP zostanie wykonane przy użyciu tokenu Bearer.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorid": [
            "3"
        ]
    },
    "id": 1
}
```

Źródło

CConnector::create() w *ui/include/classes/api/services/CConnector.php*.

connector.delete

Opis

`object connector.delete(array connectorids)`

Ta metoda umożliwia usuwanie wpisów connectora.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(tablica)` Identyfikatory konektorów do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych konektorów pod właściwością `connectorids`.

Przykłady

Usuwanie wielu konektorów

Usuń dwa wpisy konektora.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "connector.delete",
"params": [
3,
5
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"connectorids": [
"3",
"5"
]
},
"id": 1
}
```

Źródło

CConnector::delete() w *ui/include/classes/api/services/CConnector.php*.

connector.get

Opis

`integer/array connector.get(object parameters)`

Metoda umożliwia pobieranie obiektów connector zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(`object`) Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
connectorids	ID/array	Zwraca tylko connectory o podanych identyfikatorach.
selectTags	query	Zwraca właściwość `tags` z filtrem tagów connectora. Obsługuje `count`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `connectorid`, `name`, `data_type`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

(`integer/array`) Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich konektorów

Pobierz wszystkie dane o wszystkich konektorach i ich właściwościach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "connector.get",
    "params": {
        "output": "extend",
        "selectTags": ["tag", "operator", "value"],
        "preservekeys": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "connectorid": "1",
            "name": "Export of item values",
            "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": "{$DATA_EXPORT_VALUES_SSL_KEY_FILE}",
            "ssl_key_password": "",
            "description": "",
            "status": "1",
            "tags_evaltype": "0",
            "tags": [
                {
                    "tag": "component",
                    "operator": "0",
                    "value": "memory"
                }
            ]
        },
        {
            "connectorid": "2",
            "name": "Export of events",
            "protocol": "0",
            "data_type": "1",
            "url": "{$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
}
```


Źródło

CConnector:get() w *ui/include/classes/api/services/CConnector.php*.

connector.update

Opis

`object connector.update(object/array connectors)`

Ta metoda umożliwia aktualizację istniejących konektorów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości connectora do zaktualizowania.

Właściwość `connectorid` musi być zdefiniowana dla każdego connectora, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości connectora metoda akceptuje następujące parametry.

Parametr	Typ	Opis
tags	array	Filtr tagów connectora, który zastąpi bieżący filtr tagów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych konektorów we właściwości `connectorids`.

Przykłady

Zmiana typu uwierzytelniania HTTP

Zmień typ uwierzytelniania HTTP na Bearer dla connectora o ID „3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "connector.update",
    "params": {
        "connectorid": 3,
        "authtype": 5,
        "token": "{$DATA_EXPORT_BEARER_TOKEN}"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "3"
        ]
    },
    "id": 1
}
```

Aktualizowanie filtra tagów

Zmień filtr tagów dla konektora o ID „5”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "connectorids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CConnector::update() w *ui/include/classes/api/services/CConnector.php*.

Dziennik APM

Ta klasa jest przeznaczona do pracy z dziennikami zbieranymi na potrzeby monitorowania wydajności aplikacji (APM).

Odwołania do obiektów:

- Dziennik APM

Dostępne metody:

- apmlog.get - pobieranie rekordów dziennika

Obiekt dziennika APM

Poniższe obiekty są bezpośrednio powiązane z API `apmlog`.

Note:
Każda właściwość odpowiada kolumnie o tej samej nazwie w tabeli `otel_logs` źródła danych telemetrycznych (np. właściwość `severity_text` odpowiada kolumnie `SeverityText`). Listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych.

Dziennik APM

Obiekt dziennika APM reprezentuje rekord dziennika (tekstowy zapis zdarzenia, takiego jak nieudana płatność lub przekroczenie limitu czasu połączenia z bazą danych). Rekord dziennika zawiera informacje o zdarzeniu, które wystąpiło w określonym czasie, w tym informacje o jego ważności, komunikacie i atrybutach. Jeśli rekord dziennika został utworzony podczas operacji będącej częścią śladu, zawiera również identyfikatory tego śladu i odcinka.

Obiekt dziennika APM ma następujące właściwości:

Właściwość	Typ	Opis
timestamp	integer	Czas uniksowy (w nanosekundach), w którym wystąpiło zdarzenie. Zachowanie właściwości: - *tylko do odczytu*
traceid	string	Identyfikator, jeśli istnieje, śladu, podczas którego utworzono rekord dziennika. Zachowanie właściwości: - *tylko do odczytu*
spanid	string	Identyfikator, jeśli istnieje, odcinka, podczas którego utworzono rekord dziennika. Zachowanie właściwości: - *tylko do odczytu*
trace_flags	integer	Flagi śladu kontekstu śladu rekordu dziennika w postaci maski bitowej. Jeśli ustawiony jest najmniej znaczący bit (wartość 1), ślad jest próbkowany. Zachowanie właściwości: - *tylko do odczytu*
severity_text	string	Poziom ważności rekordu dziennika podany przez aplikację (np. `INFO`, `WARN` lub `ERROR`). Zachowanie właściwości: - *tylko do odczytu*
severity_number	integer	Poziom ważności rekordu dziennika jako liczba. Możliwe wartości: 0 - UNSPECIFIED; 1-4 - TRACE; 5-8 - DEBUG; 9-12 - INFO; 13-16 - WARN; 17-20 - ERROR; 21-24 - FATAL. Zachowanie właściwości: - *tylko do odczytu*
service_name	string	Nazwa usługi, która utworzyła rekord dziennika (np. `zabbix-frontend`), pobrana z atrybutu zasobu OpenTelemetry `service.name`. Zachowanie właściwości: - *tylko do odczytu*
body	string	Komunikat rekordu dziennika. Zachowanie właściwości: - *tylko do odczytu*
resource_schema_url	string	Adres URL, jeśli istnieje, schematu OpenTelemetry atrybutów zasobu. Zachowanie właściwości: - *tylko do odczytu*
resource_attributes	object	Atrybuty zasobu rekordu dziennika w postaci par klucz-wartość (np. `host.name: eabf8c2a5ddc`). Zachowanie właściwości: - *tylko do odczytu*
scope_schema_url	string	Adres URL, jeśli istnieje, schematu OpenTelemetry zakresu instrumentacji i jego atrybutów. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
scope_name	string	Nazwa zakresu instrumentacji, który utworzył rekord dziennika. Zachowanie właściwości: - *tylko do odczytu*
scope_version	string	Wersja zakresu instrumentacji, jeśli istnieje. Zachowanie właściwości: - *tylko do odczytu*
scope_attributes	object	Atrybuty zakresu instrumentacji w postaci par klucz-wartość. Zachowanie właściwości: - *tylko do odczytu*
log_attributes	object	Atrybuty dziennika rekordu dziennika w postaci par klucz-wartość (np. `log.file.path: /var/log/zabbix/zabbix_frontend.log`). Zachowanie właściwości: - *tylko do odczytu*
event_name	string	Nazwa typu zdarzenia reprezentowanego przez rekord dziennika, jeśli istnieje. Zachowanie właściwości: - *tylko do odczytu*

apmlog.get

Opis

`integer/array apmlog.get(object parameters)`

Metoda umożliwia pobieranie rekordów dziennika zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
time_from	timestamp	Zwraca tylko rekordy dziennika zdarzeń, które wystąpiły o podanym czasie lub później. Zachowanie parametru: - *wymagany*
time_till	timestamp	Zwraca tylko rekordy dziennika zdarzeń, które wystąpiły przed podanym czasem. Zachowanie parametru: - *wymagany*
traceids	string/array	Zwraca tylko rekordy dziennika utworzone podczas podanych śladów.
spanids	string/array	Zwraca tylko rekordy dziennika utworzone podczas podanych zakresów.
with_flags_on	integer	Zwraca tylko rekordy dziennika, dla których ustawione są wszystkie podane flagi śladu. Aby określić kilka flag, użyj sumy ich wartości.
with_flags_off	integer	Zwraca tylko rekordy dziennika, dla których nie jest ustawiona żadna z podanych flag śladu. Aby określić kilka flag, użyj sumy ich wartości.

Parametr	Typ	Opis
resource_attributes	object/array	Zwraca tylko rekordy dziennika z podanymi atrybutami zasobu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie rekordy dziennika. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
resource_attributes_evaltype	integer	Metoda oceny warunków w `resource_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
scope_attributes	object/array	Zwraca tylko rekordy dziennika z podanymi atrybutami zakresu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie rekordy dziennika. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
scope_attributes_evaltype	integer	Metoda oceny warunków w `scope_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
log_attributes	object/array	Zwraca tylko rekordy dziennika z podanymi atrybutami dziennika. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie rekordy dziennika. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
log_attributes_evaltype	integer	Metoda oceny warunków w `log_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
filter	object	Zwraca tylko wyniki dokładnie pasujące do podanego filtra. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncze wartości lub tablice wartości, do których należy dopasować wyniki. Obsługiwane właściwości: `traceid`, `spanid`, `trace_flags`, `severity_text`, `severity_number`, `service_name`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `event_name`.

Parametr	Typ	Opis
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są ciągi znaków, których należy szukać. Jeśli nie podano dodatkowych opcji, wyszukiwanie będzie wykonywane za pomocą `LIKE "%…%"`. Obsługiwane właściwości: `severity_text`, `service_name`, `body`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `event_name`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `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	Liczba rekordów pomijanych przed zwróceniem wyniku.
countOutput	boolean	Parametry te opisano w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Wartości zwracane

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie dzienników błędów śladu

Pobierz rekordy dziennika o ważności ERROR (numery ważności 17–20) powiązane ze śladem „8a4f1c7e2b9d4e6fa0c3b5d7e9f1a2c4”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```


Wyszukiwanie komunikatów w dzienniku

Pobierz czas, usługę i komunikat 10 najnowszych rekordów dziennika, których komunikat zawiera tekst „timeout”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:


```
{
    "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
}
```

Zobacz także

- Ślad APM
- Zakres APM
- Metryka APM

Źródło

CApmLog::get() w *ui/include/classes/api/services/CApmLog.php*.

Dziennik audytu

Ta klasa została zaprojektowana do pracy z dziennikiem audytu.

Odwołania do obiektów:

- Dziennik audytu

Dostępne metody:

- auditlog.get - pobieranie rekordów dziennika audytu

Obiekt dziennika audytu

Następujące obiekty są bezpośrednio powiązane z API `auditlog`.

Dziennik audytu

Obiekt dziennika audytu zawiera informacje o działaniach użytkownika. Ma następujące właściwości.

Właściwość	Typ	Opis
auditid	ID	ID wpisu dziennika audytu. Generowane przy użyciu algorytmu CUID.
userid	ID	Userid autora wpisu dziennika audytu.
username	string	Nazwa użytkownika autora wpisu dziennika audytu.
clock	timestamp	Znacznik czasu utworzenia wpisu dziennika audytu.
ip	string	Adres IP autora wpisu dziennika audytu.
action	integer	Działanie wpisu dziennika audytu. Możliwe wartości: 0 - Dodanie; 1 - Aktualizacja; 2 - Usunięcie; 4 - Wylogowanie; 7 - Wykonanie; 8 - Logowanie; 9 - Nieudane logowanie; 10 - Czyszczenie historii; 11 - Odświeżenie konfiguracji; 12 - Push.

Właściwość	Typ	Opis
resourcetype	integer	Typ zasobu wpisu dziennika audytu. Możliwe wartości: 0 - Użytkownik; 3 - Typ nośnika; 4 - Host; 5 - Akcja; 6 - Wykres; 11 - Grupa użytkowników; 13 - Wyzwalacz; 14 - Grupa hostów; 15 - Pozycja; 16 - Obraz; 17 - Mapa wartości; 18 - Usługa; 19 - Mapa; 22 - Scenariusz WWW; 23 - Reguła wykrywania; 25 - Skrypt; 26 - Proxy; 27 - Okres utrzymania; 28 - Wyrażenie regularne; 29 - Makro; 30 - Szablon; 31 - Prototyp wyzwalacza; 32 - Mapowanie ikon; 33 - Pulpit; 34 - Korelacja zdarzeń; 35 - Prototyp wykresu; 36 - Prototyp pozycji; 37 - Prototyp hosta; 38 - Autorejestracja; 39 - Moduł; 40 - Ustawienia; 41 - Housekeeping; 42 - Uwierzytelnianie; 43 - Pulpit szablonu; 44 - Rola użytkownika; 45 - Token API; 46 - Zaplanowany raport; 47 - Węzeł wysokiej dostępności; 48 - SLA; 49 - Katalog użytkowników; 50 - Grupa szablonów; 51 - Łącznik; 52 - Reguła LLD; 53 - Historia; 54 - Uwierzytelnianie wieloskładnikowe; 55 - Grupa proxy; 56 - Prototyp reguły LLD.
resourceid	ID	Identyfikator zasobu wpisu dziennika audytu.
resource_cuid	ID	Unikalny identyfikator zasobu wpisu dziennika audytu wygenerowany przy użyciu algorytmu CUID.
resourcename	string	Czytelna dla człowieka nazwa zasobu wpisu dziennika audytu.
recordsetid	ID	ID zestawu rekordów wpisu dziennika audytu. Rekordy dziennika audytu utworzone podczas tej samej operacji będą miały to samo ID zestawu rekordów. Generowane przy użyciu algorytmu CUID.

Właściwość	Typ	Opis
details	text	Szczegóły wpisu dziennika audytu. Szczegóły są przechowywane jako obiekt JSON, gdzie każda nazwa właściwości jest ścieżką do właściwości lub zagnieżdżonego obiektu, w którym wystąpiła zmiana, a każda wartość zawiera dane (w formacie tablicy) o zmianie tej właściwości lub zagnieżdżonego obiektu. Możliwe formaty wartości: ["add"] - Zagnieżdżony obiekt został dodany; ["add", "<value>"] - Właściwość dodanego obiektu jest równa <value>; ["update"] - Zagnieżdżony obiekt został zaktualizowany; ["update", "<new value>", "<old value>"] - Właściwość zaktualizowanego obiektu została zmieniona z <old value> na <new value>; ["delete"] - Zagnieżdżony obiekt został usunięty.

auditlog.get

Opis

`integer/array auditlog.get(object parameters)`

Metoda umożliwia pobieranie rekordów dziennika audytu zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
auditids	ID/array	Zwraca tylko dziennik audytu o podanych ID.
userids	ID/array	Zwraca tylko dziennik audytu utworzony przez podanych użytkowników.
time_from	timestamp	Zwraca tylko wpisy dziennika audytu, które zostały utworzone po podanym czasie lub o podanym czasie.
time_till	timestamp	Zwraca tylko wpisy dziennika audytu, które zostały utworzone przed podanym czasem lub o podanym czasie.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `auditid`, `userid`, `clock`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;

- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie dziennika audytu

Pobierz dwa najnowsze wpisy dziennika audytu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "auditlog.get",
    "params": {
        "output": "extend",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 2
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Audit log object

Źródło

CAuditLog::get() in *ui/include/classes/api/services/CAuditLog.php*.

Grupa hostów

Ta klasa jest przeznaczona do pracy z grupami hostów.

Odwołania do obiektów:

- Grupa hostów

Dostępne metody:

- hostgroup.create - utwórz nowe grupy hostów
- hostgroup.delete - usuń grupy hostów
- hostgroup.get - pobierz grupy hostów
- hostgroup.massadd - dodaj powiązane obiekty do grup hostów
- hostgroup.massremove - usuń powiązane obiekty z grup hostów
- hostgroup.propagate - propaguj uprawnienia i filtry tagów do podgrup grup hostów
- hostgroup.update - zaktualizuj grupy hostów

Obiekt grupy hostów

Następujące obiekty są bezpośrednio powiązane z API `hostgroup`.

Grupa hostów

Obiekt grupy hostów ma następujące właściwości.

Właściwość	Typ	Opis
groupid	ID	ID grupy hostów. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa grupy hostów. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie grupy hostów. Możliwe wartości: 0 - zwykła grupa hostów; 4 - wykryta grupa hostów. Zachowanie właściwości: - *tylko do odczytu*
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych grup hostów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.

hostgroup.create

Opis

`object hostgroup.create(object/array hostGroups)`

Ta metoda umożliwia tworzenie nowych grup hostów.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Grupy hostów do utworzenia.

Metoda akceptuje grupy hostów z standardowymi właściwościami grup hostów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup hostów we właściwości `groupids`. Kolejność zwróconych ID odpowiada kolejności przekazanych grup hostów.

Przykłady

Tworzenie grupy hostów

Utwórz grupę hostów o nazwie „Serwery Linux”.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostgroup.create",
"params": {
"name": "Serwery Linux"
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"groupids": [
"107819"
]
},
"id": 1
}
```

Źródło

CHostGroup::create() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.delete

Opis

`object hostgroup.delete(array hostGroupIds)`

Ta metoda umożliwia usuwanie grup hostów.

Grupy hostów nie można usunąć, jeśli:

- zawiera hosty, które należą wyłącznie do tej grupy;
- jest oznaczona jako wewnętrzna;
- jest używana przez prototyp hosta;
- jest używana w skrypcie globalnym;
- jest używana w warunku korelacji.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID grup hostów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup hostów w właściwości `groupids`.

Przykłady

Usuwanie wielu grup hostów

Usuń dwie grupy hostów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostgroup.delete",
"params": [
"107824",
"107825"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"groupids": [
"107824",
"107825"
]
},
"id": 1
}
```

Źródło

CHostGroup::delete() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.get

Opis

`integer/array hostgroup.get(object parameters)`

Metoda umożliwia pobieranie grup hostów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwraca tylko grupy hostów zawierające hosty z podanymi wykresami.
groupids	ID/array	Zwraca tylko grupy hostów o podanych identyfikatorach grup hostów.
hostids	ID/array	Zwraca tylko grupy hostów zawierające podane hosty.
maintenanceids	ID/array	Zwraca tylko grupy hostów, na które wpływają podane okresy konserwacji.
triggerids	ID/array	Zwraca tylko grupy hostów zawierające hosty z podanymi wyzwalaczami.
with_graphs	boolean	Zwraca tylko grupy hostów zawierające hosty z wykresami.
with_graph_prototypes	boolean	Zwraca tylko grupy hostów zawierające hosty z prototypami wykresów.
with_hosts	boolean	Zwraca tylko grupy hostów zawierające hosty.
with_httptests	boolean	Zwraca tylko grupy hostów zawierające hosty z testami WWW. Nadpisuje parametr `with_monitored_httptests`.
with_items	boolean	Zwraca tylko grupy hostów zawierające hosty z pozycjami. Nadpisuje parametry `with_monitored_items` i `with_simple_graph_items`.

Parametr	Typ	Opis
with_item_prototypes	boolean	Zwraca tylko grupy hostów zawierające hosty z prototypami pozycji. Nadpisuje parametr `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	boolean	Zwraca tylko grupy hostów zawierające hosty z prototypami pozycji, które są włączone do tworzenia i mają numeryczny typ informacji.
with_monitored_httptests	boolean	Zwraca tylko grupy hostów zawierające hosty z włączonymi testami WWW.
with_monitored_hosts	boolean	Zwraca tylko grupy hostów zawierające monitorowane hosty.
with_monitored_items	boolean	Zwraca tylko grupy hostów zawierające hosty z włączonymi pozycjami. Nadpisuje parametr `with_simple_graph_items`.
with_monitored_triggers	boolean	Zwraca tylko grupy hostów zawierające hosty z włączonymi wyzwalaczami. Wszystkie pozycje używane w wyzwalaczu również muszą być włączone.
with_simple_graph_items	boolean	Zwraca tylko grupy hostów zawierające hosty z numerycznymi pozycjami.
with_triggers	boolean	Zwraca tylko grupy hostów zawierające hosty z wyzwalaczami. Nadpisuje parametr `with_monitored_triggers`.
selectDiscoveryRules	query	Zwraca właściwość `discoveryRules` z regułami LLD, które wykryły grupę hostów.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z obiektami wykrywania grup hostów. Każdy obiekt wykrywania grupy hostów jest prototypem grupy hostów powiązanym z wykrytą grupą hostów i ma następujące właściwości: `parent_group_prototypeid` - `(ID)` identyfikator prototypu grupy hostów, z którego wykryto grupę hostów; `name` - `(string)` nazwa prototypu grupy hostów; `ts_delete` - `(timestamp)` czas, po którym grupa hostów, która nie jest już wykrywana, zostanie usunięta; `status` - `(int)` status wykrywania grupy hostów: 0 - *(domyślnie)* grupa hostów jest wykryta, 1 - grupa hostów nie jest już wykrywana.
selectHostPrototypes	query	Zwraca właściwość `hostPrototypes` z prototypami hostów, które wykryły tę grupę hostów.
selectHosts	query	Zwraca właściwość `hosts` z hostami należącymi do grupy hostów. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wyniki będą sortowane według `host`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `groupid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Parametr	Typ	Opis
selectGroupDiscoveries	query	Zwraca właściwość `groupDiscoveries` z obiektami wykrywania grup hostów. Każdy obiekt wykrywania grupy hostów jest prototypem grupy hostów powiązanym z wykrytą grupą hostów. To zapytanie jest **przestarzałe**; zamiast niego użyj `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`

Przykłady

Pobieranie danych według nazwy

Pobierz wszystkie dane o dwóch grupach hostów o nazwach „Zabbix servers” i „Linux servers”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Zabbix servers",
                "Linux servers"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "2",
            "name": "Linux servers",
            "flags": "0",
            "uuid": "dc579cd7a1a34222933f24f52a68bcd8"
        },
        {
            "groupid": "4",
            "name": "Zabbix servers",
            "flags": "0",
            "uuid": "6f6799aa69e844b4b3918f779f2abf08"
        }
    ],
    "id": 1
}
```

Zobacz także

- Host

Źródło

CHostGroup::get() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massadd

Opis

`object hostgroup.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wszystkich podanych grup hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup hostów do zaktualizowania oraz obiekty do dodania do wszystkich grup hostów.

Metoda akceptuje następujące parametry.

Parametr	Type	Opis
groups	object/array	Grupy hostów, które mają zostać zaktualizowane. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametrów: - *wymagane*
hosts	object/array	Hosty do dodania do wszystkich grup hostów. Hosty muszą mieć zdefiniowaną tylko właściwość `hostid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów we właściwości `groupids`.

Przykłady

Dodawanie hostów do grup hostów

Dodaj dwa hosty do grup hostów o identyfikatorach 5 i 6.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "5"
            },
            {
                "groupid": "6"
            }
        ],
        "hosts": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30001"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host

Źródło

CHostGroup::massAdd() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.massremove

Opis

`object hostgroup.massremove(object parameters)`

Ta metoda umożliwia usuwanie powiązanych obiektów z wielu grup hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup hostów do zaktualizowania oraz obiekty, które powinny zostać usunięte.

Parametr	Type	Opis
groupids	ID/array	Identyfikatory grup hostów, które mają zostać zaktualizowane. Zachowanie parametru: - *wymagany*
hostids	ID/array	Identyfikatory hostów, które mają zostać usunięte ze wszystkich grup hostów.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów we właściwości `groupids`.

Przykłady

Usuwanie hostów z grup hostów

Usuń dwa hosty z podanych grup hostów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
        "hostids": [
            "30050",
```

```
            "30001"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CHostGroup::massRemove() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.propagate

Opis

`object hostgroup.propagate(object parameters)`

Ta metoda umożliwia zastosowanie uprawnień i filtrów tagów do wszystkich podgrup host group.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

|Parametr|Type|Opis|

|--|--|------|

|groups|object/array|Grupy hostów do propagowania.

Grupy hostów muszą mieć zdefiniowaną właściwość `groupid`.

Zach
parametrów:
- *wymagane*|

|permissions|boolean|Ustaw na ”true”, aby propagować uprawnienia.

Zachowanie parametrów:
- *wymagane*, jeśli `tag_filters` nie jest ustawione|

|tag_filters|boolean|Ustaw na ”true”, aby propagować filtry tagów.

Zachowanie parametrów:
- *wymagane*, jeśli `permissions` nie jest ustawione|

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory propagowanych grup hostów

we właściwości `groupids`.

Przykłady

Propagowanie uprawnień grupy hostów i filtrów tagów do jej podgrup.

Propagowanie uprawnień grupy hostów i filtrów tagów do jej podgrup.

Żądanie:

```
{

    "jsonrpc": "2.0",
```

```
    "method": "hostgroup.propagate",
    "params": {
        "groups": [
            {
                "groupid": "6"
            }
        ],
        "permissions": true,
        "tag_filters": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "6",
        ]
    },
    "id": 1
}
```

Zobacz także

- hostgroup.update
- hostgroup.massadd
- Host

Źródło

CHostGroup::propagate() w *ui/include/classes/api/services/CHostGroup.php*.

hostgroup.update

Opis

`object hostgroup.update(object/array hostGroups)`

Ta metoda umożliwia aktualizację istniejących grup hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości grupy hostów, które mają zostać zaktualizowane.

Właściwość `groupid` musi być zdefiniowana dla każdej grupy hostów, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup hostów we właściwości `groupids`.

Przykłady

Zmiana nazwy grupy hostów

Zmień nazwę grupy hostów na „Linux hosts”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.update",
    "params": {
        "groupid": "7",
        "name": "Linux hosts"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Zmiana nazw wielu grup hostów

Zmień nazwy dwóch grup hostów, aby uwzględnić w nich informacje o regionie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostgroup.update",
    "params": [
        {
            "groupid": "8",
            "name": "Europe/Linux hosts"
        },
        {
            "groupid": "9",
            "name": "Europe/Windows hosts"
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "8",
            "9"
        ]
    },
    "id": 1
}
```

Źródło

CHostGroup::update() w *ui/include/classes/api/services/CHostGroup.php*.

Grupa proxy

Ta klasa jest przeznaczona do pracy z grupami proxy.

Odwołania do obiektów:

- Grupa proxy

Dostępne metody:

- proxygroup.create - utwórz nowe grupy proxy
- proxygroup.delete - usuń grupy proxy
- proxygroup.get - pobierz grupy proxy
- proxygroup.update - zaktualizuj grupy proxy

Obiekt grupy proxy

Następujące obiekty są bezpośrednio powiązane z API `proxygroup`.

Grupa proxy

Obiekt grupy proxy ma następujące właściwości.

Property	Type	Description
proxy_groupid	ID	Identyfikator grupy proxy. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
name	string	Nazwa grupy proxy. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
description	text	Opis grupy proxy.
failover_delay	string	Okres, w którym proxy w grupie proxy musi komunikować się z serwerem Zabbix, aby zostać uznane za dostępne. Obsługiwane są przyrostki czasu (np. 30s, 1m). Obsługiwane są makra użytkownika. Możliwe wartości: 10s-15m. Wartość domyślna: 1m.

Property	Type	Description
min_online	string	Minimalna liczba dostępnych proxy wymagana do utrzymania dostępności grupy proxy. Obsługiwane są makra użytkownika. Zakres możliwych wartości: 1-1000. Wartość domyślna: 1.
state	integer	Stan grupy proxy. Możliwe wartości: 0 - Nieznany; 1 - Niedostępna; 2 - Odzyskiwanie; 3 - Dostępna; 4 - Pogarszający się. Zachowanie właściwości: *- tylko do odczytu*

proxygroup.create

Opis

`object proxygroup.create(object/array proxyGroups)`

Ta metoda umożliwia tworzenie nowych grup proxy.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Grupy proxy do utworzenia.

Metoda akceptuje grupy proxy ze standardowymi właściwościami grup proxy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup proxy we właściwości `proxy_groupids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych grup proxy.

Przykłady

Utwórz grupę proxy

Utwórz grupę proxy z niestandardowymi ustawieniami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.create",
    "params": {
        "name": "Proxy group",
        "failover_delay": "5m",
        "min_online": "10"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CProxyGroup::create() w *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.delete

Opis

`object proxygroup.delete(array proxyGroupIds)`

Ta metoda umożliwia usuwanie grup proxy.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID grup proxy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup proxy we właściwości `proxy_groupids`.

Przykłady

Usuń wiele grup proxy

Usuń dwie grupy proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.delete",
    "params": [
        "5",
        "10"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5",
            "10"
        ]
    },
    "id": 1
}
```

Źródło

CProxyGroup::delete() w *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.get

Opis

`integer/array proxygroup.get(object parameters)`

Metoda umożliwia pobieranie grup proxy zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników. Zwracane są tylko grupy proxy dozwolone przez *listę dostępu proxy* w grupach użytkowników danego użytkownika.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
proxy_groupids	ID/array	Zwraca tylko grupy proxy o podanych identyfikatorach.
proxyids	ID/array	Zwraca tylko grupy proxy zawierające podane proxy.
selectProxies	query	Zwraca właściwość `proxies` z proxy należącymi do grupy proxy. Obsługuje `count`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `proxy_groupid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich grup proxy

Pobierz wszystkie skonfigurowane grupy proxy wraz z proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.get",
    "params": {
        "output": "extend",
        "selectProxies": ["proxyid", "name"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Proxy

Źródło

CProxyGroup::get() w *ui/include/classes/api/services/CProxyGroup.php*.

proxygroup.update

Opis

`object proxygroup.update(object/array proxyGroups)`

Ta metoda umożliwia aktualizację istniejących grup proxy.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości grupy proxy do zaktualizowania.

Właściwość `proxy_groupid` musi być zdefiniowana dla każdej grupy proxy, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje grupy proxy ze standardowymi właściwościami grupy proxy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup proxy we właściwości `proxy_groupids`.

Przykłady

Zmień minimalną liczbę proxy online

Zmień minimalną liczbę proxy online wymaganą, aby grupa była online.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxygroup.update",
    "params": {
        "proxy_groupid": "5",
        "min_online": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxy_groupids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CProxyGroup::update() w *ui/include/classes/api/services/CProxyGroup.php*.

Grupa szablonów

Ta klasa jest przeznaczona do pracy z grupami szablonów.

Odwołania do obiektów:

- Grupa szablonów

Dostępne metody:

- templategroup.create - utwórz nowe grupy szablonów
- templategroup.delete - usuń grupy szablonów
- templategroup.get - pobierz grupy szablonów
- templategroup.massadd - dodaj powiązane obiekty do grup szablonów
- templategroup.massremove - usuń powiązane obiekty z grup szablonów
- templategroup.propagate - propaguj uprawnienia do podgrup grup szablonów
- templategroup.update - zaktualizuj grupy szablonów

Obiekt grupy szablonów

Następujące obiekty są bezpośrednio powiązane z API `templategroup`.

Grupa szablonów

Obiekt grupy szablonów ma następujące właściwości.

Właściwość	Typ	Opis
groupid	ID	ID grupy szablonów. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa grupy szablonów. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych grup szablonów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.

templategroup.create

Opis

`object templategroup.create(object/array templateGroups)`

Ta metoda umożliwia tworzenie nowych grup szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Grupy szablonów do utworzenia. Metoda akceptuje grupy szablonów ze standardowymi właściwościami grup szablonów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup szablonów we właściwości `groupids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych grup szablonów.

Przykłady

Tworzenie grupy szablonów

Utwórz grupę szablonów o nazwie "Templates/Databases".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.create",
    "params": {
        "name": "Templates/Databases"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
```

```
            "107820"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::create() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.delete

Opis

`object templategroup.delete(array templateGroupIds)`

Ta metoda umożliwia usuwanie grup szablonów.

Grupy szablonów nie można usunąć, jeśli zawiera szablony należące wyłącznie do tej grupy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID grup szablonów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup szablonów we właściwości `groupids`.

Przykłady

Usuwanie wielu grup szablonów

Usuń dwie grupy szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.delete",
    "params": [
        "107814",
        "107815"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "107814",
            "107815"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::delete() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.get

Opis

`integer/array templategroup.get(object parameters)`

Metoda umożliwia pobieranie grup szablonów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwracaj tylko grupy szablonów zawierające szablony z podanymi wykresami.
groupids	ID/array	Zwracaj tylko grupy szablonów o podanych identyfikatorach grup szablonów.
templateids	ID/array	Zwracaj tylko grupy szablonów zawierające podane szablony.
triggerids	ID/array	Zwracaj tylko grupy szablonów zawierające szablony z podanymi wyzwalaczami.
with_graphs	boolean	Zwracaj tylko grupy szablonów zawierające szablony z wykresami.
with_graph_prototypes	boolean	Zwracaj tylko grupy szablonów zawierające szablony z prototypami wykresów.
with_httptests	boolean	Zwracaj tylko grupy szablonów zawierające szablony z testami WWW.
with_items	boolean	Zwracaj tylko grupy szablonów zawierające szablony z pozycjami. Nadpisuje parametry `with_simple_graph_items`.
with_item_prototypes	boolean	Zwracaj tylko grupy szablonów zawierające szablony z prototypami pozycji. Nadpisuje parametr `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	boolean	Zwracaj tylko grupy szablonów zawierające szablony z prototypami pozycji, które są włączone do tworzenia i mają numeryczny typ informacji.
with_simple_graph_items	boolean	Zwracaj tylko grupy szablonów zawierające szablony z numerycznymi pozycjami.
with_templates	boolean	Zwracaj tylko grupy szablonów zawierające szablony.
with_triggers	boolean	Zwracaj tylko grupy szablonów zawierające szablony z wyzwalaczami.
selectTemplates	query	Zwracaj właściwość `templates` zawierającą szablony należące do grupy szablonów. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectTemplates` - wyniki będą sortowane według `template`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `groupid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	

Parametr	Typ	Opis
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych według nazwy

Pobierz wszystkie dane o dwóch grupach szablonów o nazwach „Templates/Databases” oraz „Templates/Modules”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.get",
    "params": {
        "output": "extend",
        "filter": {
            "name": [
                "Templates/Databases",
                "Templates/Modules"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "groupid": "13",
            "name": "Templates/Databases",
            "uuid": "748ad4d098d447d492bb935c907f652f"
        },
        {
            "groupid": "8",
            "name": "Templates/Modules",
            "uuid": "57b7ae836ca64446ba2c296389c009b7"
        }
    ],
    "id": 1
}
```

Zobacz także

- Szablon

Źródło

CTemplateGroup::get() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massadd

Opis

`object templategroup.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wszystkich podanych grup szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup szablonów do zaktualizowania oraz obiekty do dodania do wszystkich grup szablonów.

Metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy szablonów, które mają zostać zaktualizowane. Grupy szablonów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`. Zachowanie parametrów: - *wymagane*
templates	object/array	Szablony do dodania do wszystkich grup szablonów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`. Zachowanie parametrów: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów we właściwości `groupids`.

Przykłady

Dodawanie szablonów do grup szablonów

Dodaj dwa szablony do grup szablonów o identyfikatorach 12 i 13.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "12"
            },
            {
                "groupid": "13"
            }
        ],
        "templates": [
            {
                "templateid": "10486"
            },
            {
                "templateid": "10487"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "12",
            "13"
        ]
    },
    "id": 1
}
```

Zobacz także

- Szablon

Źródło

CTemplateGroup::massAdd() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.massremove

Opis

`object templategroup.massremove(object parameters)`

Ta metoda umożliwia usunięcie powiązanych obiektów z wielu grup szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory grup szablonów do zaktualizowania oraz obiekty, które powinny zostać usunięte.

Parametr	Typ	Opis
groupids	ID/array	Identyfikatory grup szablonów, które mają zostać zaktualizowane. Zachowanie parametrów: - *wymagane*
templateids	ID/array	Identyfikatory szablonów, które mają zostać usunięte ze wszystkich grup szablonów. Zachowanie parametrów: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów we właściwości `groupids`.

Przykłady

Usuwanie szablonów z grup szablonów

Usuń dwa szablony z podanych grup szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.massremove",
    "params": {
        "groupids": [
            "5",
            "6"
        ],
```

```
        "templateids": [
            "30050",
            "30001"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::massRemove() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.propagate

Opis

`object templategroup.propagate(object parameters)`

Ta metoda umożliwia zastosowanie uprawnień do wszystkich podgrup grup szablonów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typów użytkowników *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

|Parametr|Type|Opis|

|--|--|------|

|groups|object/array|Grupy szablonów do propagowania.

Grupy szablonów muszą mieć zdefiniowaną tylko właściwość `groupid`.

Zachowanie parametrów:
- *wymagane*|

|permissions|boolean|Ustaw `true`, jeśli należy propagować uprawnienia.

Zachowanie parametrów:
- *wymagane*|

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory propagowanych grup szablonów

we właściwości `groupids`.

Przykłady

Propagowanie uprawnień grupy szablonów do jej podgrup.

Propaguj uprawnienia grupy szablonów do jej podgrup.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.propagate",
    "params": {
        "groups": [
```

```
            {
                "groupid": "15"
            }
        ],
        "permissions": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "15",
        ]
    },
    "id": 1
}
```

Zobacz też

- templategroup.update
- templategroup.massadd
- Szablon

Źródło

CTemplateGroup::propagate() w *ui/include/classes/api/services/CTemplateGroup.php*.

templategroup.update

Opis

`object templategroup.update(object/array templateGroups)`

Ta metoda umożliwia aktualizację istniejących grup szablonów.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości grupy szablonów, które mają zostać zaktualizowane.

Właściwość `groupid` musi być zdefiniowana dla każdej grupy szablonów, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup szablonów we właściwości `groupids`.

Przykłady

Zmiana nazwy grupy szablonów

Zmień nazwę grupy szablonów na ”Templates/Databases”

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templategroup.update",
    "params": {
        "groupid": "7",
        "name": "Templates/Databases"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "7"
        ]
    },
    "id": 1
}
```

Źródło

CTemplateGroup::update() w *ui/include/classes/api/services/CTemplateGroup.php*.

Grupa użytkowników

Ta klasa została zaprojektowana do pracy z grupami użytkowników.

Odwołania do obiektów:

- Grupa użytkowników
- Uprawnienie
- Uprawnienie oparte na tagach

Dostępne metody:

- usergroup.create - utwórz nowe grupy użytkowników
- usergroup.delete - usuń grupy użytkowników
- usergroup.get - pobierz grupy użytkowników
- usergroup.update - zaktualizuj grupy użytkowników

Obiekt grupy użytkowników

Następujące obiekty są bezpośrednio powiązane z API `usergroup`.

Grupa użytkowników

Obiekt grupy użytkowników ma następujące właściwości.

Property	Type	Description
usrgrpid	ID	Identyfikator grupy użytkowników. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* w operacjach aktualizacji
name	string	Nazwa grupy użytkowników. Zachowanie właściwości: - *wymagana* w operacjach tworzenia
debug_mode	integer	Określa, czy tryb debugowania jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* wyłączony; 1 - włączony.

Property	Type	Description
gui_access	integer	Metoda uwierzytelniania frontend użytkowników w grupie. Możliwe wartości: 0 - *(domyślnie)* użyj domyślnej systemowej metody uwierzytelniania; 1 - użyj uwierzytelniania wewnętrznego; 2 - użyj uwierzytelniania LDAP; 3 - wyłącz dostęp do frontend.
mfa_status	integer	Określa, czy uwierzytelnianie MFA jest włączone, czy wyłączone dla użytkowników w grupie. Możliwe wartości: 0 - wyłączone (dla wszystkich skonfigurowanych metod MFA); 1 - włączone (dla wszystkich skonfigurowanych metod MFA).
mfaid	ID	Metoda MFA używana przez użytkowników w grupie. Zachowanie właściwości: - *obsługiwana* dla użytkowników typu *Super admin* - *obsługiwana*, jeśli parametr `mfa_status` w obiekcie uwierzytelniania jest ustawiony na „Włączone”
proxy_mode	integer	Tryb obsługi dostępu do proxy, które nie są uwzględnione w grupach proxy. Możliwe wartości: 0 - lista odmów; 1 - *(domyślnie)* lista zezwoleń. Zachowanie właściwości: - *obsługiwana* dla użytkowników typu *Super admin*.
proxy_group_mode	integer	Tryb obsługi dostępu do grup proxy (i ich proxy). Możliwe wartości: 0 - lista odmów; 1 - *(domyślnie)* lista zezwoleń. Zachowanie właściwości: - *obsługiwana* dla użytkowników typu *Super admin*.
users_status	integer	Określa, czy grupa użytkowników jest włączona, czy wyłączona. Dla użytkowników wycofanych z obsługi nie można włączyć grupy użytkowników. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.
userdirectoryid	ID	Identyfikator katalogu użytkowników używanego do uwierzytelniania. Zachowanie właściwości: - *obsługiwana* dla użytkowników typu *Super admin* - *obsługiwana*, jeśli `gui_access` jest ustawiony na „użyj domyślnej systemowej metody uwierzytelniania” lub „użyj uwierzytelniania LDAP”

Uprawnienie

Obiekt uprawnienia ma następujące właściwości.

Właściwość	Typ	Opis
id	ID	ID grupy hostów lub grupy szablonów, do której ma zostać dodane uprawnienie. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
permission	integer	Poziom dostępu do grupy hostów lub grupy szablonów. Możliwe wartości: 0 - dostęp zabroniony; 2 - dostęp tylko do odczytu; 3 - dostęp do odczytu i zapisu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Uprawnienie oparte na tagach

Obiekt uprawnienia opartego na tagach ma następujące właściwości.

Właściwość	Typ	Opis
groupid	ID	ID grupy hostów, do której należy dodać uprawnienie. Zachowanie właściwości: - *wymagane*
tag	string	Nazwa tagu.
value	string	Wartość tagu.

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.
signal_type	integer	APM resource attribute category. Possible values: -1 - *(default)* All; 0 - Traces; 1 - Metrics; 2 - Logs.

usergroup.create

Opis

`object usergroup.create(object/array userGroups)`

Ta metoda umożliwia tworzenie nowych grup użytkowników.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Grupy użytkowników do utworzenia.

Oprócz standardowych właściwości grupy użytkowników metoda przyjmuje następujące parametry.

Parametr	Typ	Opis
hostgroup_rights	object/array	Uprawnienia do grup hostów, które zostaną przypisane do grupy użytkowników.
proxies	object/array	Proxy, które zostaną powiązane z grupą użytkowników. Proxy muszą mieć zdefiniowaną wyłącznie właściwość `proxyid`.
proxy_groups	object/array	Grupy proxy, które zostaną powiązane z grupą użytkowników. Grupy proxy muszą mieć zdefiniowaną wyłącznie właściwość `proxy_groupid`.
templategroup_rights	object/array	Uprawnienia do grup szablonów, które zostaną przypisane do grupy użytkowników.
tag_filters	array	Uprawnienia oparte na tagach, które zostaną przypisane do grupy użytkowników.
users	object/array	Użytkownicy, którzy zostaną dodani do grupy użytkowników. Użytkownik musi mieć zdefiniowaną wyłącznie właściwość `userid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych grup użytkowników we właściwości `usrgrpids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych grup użytkowników.

Przykłady

Tworzenie grupy użytkowników

Utwórz grupę użytkowników *Operation managers* z zabronionym dostępem do grupy hostów „2” i dodaj do niej użytkownika.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.create",
    "params": {
        "name": "Operation managers",
        "hostgroup_rights": {
            "id": "2",
            "permission": 0
        },
        "users": [
            {
                "userid": "12"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "20"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uprawnienie

Źródło

CUserGroup::create() w *ui/include/classes/api/services/CUserGroup.php*.

usergroup.delete

Opis

`object usergroup.delete(array userGroupIds)`

Ta metoda umożliwia usuwanie grup użytkowników.

> **Attention:**
> Grupa użytkowników Deprovisioned (grupa użytkowników określona dla `disabled_usrgrpid` w Authentication) nie może zostać usunięta.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(array)` ID grup użytkowników do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych grup użytkowników we właściwości `usrgrpids`.

Przykłady

Usuwanie wielu grup użytkowników

Usuń dwie grupy użytkowników.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "usergroup.delete",
"params": [
"20",
"21"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"usrgrpids": [
"20",
"21"
]
},
"id": 1
}
```

Źródło

CUserGroup::delete() w *ui/include/classes/api/services/CUserGroup.php*.

usergroup.get

Opis

`integer/array usergroup.get(object parameters)`

Metoda umożliwia pobranie grup użytkowników zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
mfaids	ID/array	Zwraca tylko grupy użytkowników z podanymi metodami MFA. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*
mfa_status	integer	Zwraca tylko grupy użytkowników z podanym statusem MFA. Listę obsługiwanych statusów można znaleźć na stronie grup użytkowników.
proxyids	ID/array	Zwraca tylko grupy użytkowników zawierające podane proxy. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*.
proxy_groupids	ID/array	Zwraca tylko grupy użytkowników zawierające podane grupy proxy. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*.
selectProxies	ID/array	Zwraca właściwość `proxies` zawierającą proxy grup użytkowników. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*.
selectProxyGroups	ID/array	Zwraca właściwość `proxy_groups` zawierającą grupy proxy grup użytkowników. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*.
status	integer	Zwraca tylko grupy użytkowników z podanym statusem. Listę obsługiwanych statusów można znaleźć na stronie grup użytkowników.
userids	ID/array	Zwraca tylko grupy użytkowników zawierające podanych użytkowników.
usrgrpids	ID/array	Zwraca tylko grupy użytkowników o podanych identyfikatorach.
selectTagFilters	query	Zwraca uprawnienia grup użytkowników oparte na tagach we właściwości `tag_filters`.
selectUsers	query	Zwraca użytkowników z grupy użytkowników we właściwości `users`. Ograniczenia dotyczące typu użytkownika opisano w `user.get`.
selectHostGroupRights	query	Zwraca uprawnienia grup użytkowników do grup hostów we właściwości `hostgroup_rights`. Listę poziomów dostępu do grup hostów można znaleźć na stronie grup użytkowników.
selectTemplateGroupRights	query	Zwraca uprawnienia grup użytkowników do grup szablonów we właściwości `templategroup_rights`. Listę poziomów dostępu do grup szablonów można znaleźć na stronie grup użytkowników.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania.

Parametr	Typ	Opis
output	query	Właściwości obiektu grupy użytkowników, które mają zostać zwrócone. Użytkownicy typu *Admin* i *User* mogą pobierać tylko następujące właściwości: `usrgrpid`, `name`, `gui_access`, `users_status`, `debug_mode`, `mfa_status`. Wartość domyślna: `extend`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `usrgrpid`, `name`.
countOutput	boolean	Opis tych parametrów znajduje się w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie włączonych grup użytkowników

Pobierz wszystkie włączone grupy użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.get",
    "params": {
        "output": "extend",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Użytkownik

Źródło

CUserGroup::get() w *ui/include/classes/api/services/CUserGroup.php*.

usergroup.update

Opis

`object usergroup.update(object/array userGroups)`

Ta metoda umożliwia aktualizację istniejących grup użytkowników.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości grup użytkowników, które mają zostać zaktualizowane.

Właściwość `usrgrpid` musi być zdefiniowana dla każdej grupy użytkowników, a wszystkie pozostałe właściwości są opcjonalne. Zostaną zaktualizowane tylko przekazane właściwości, a wszystkie pozostałe pozostaną niezmienione.

Oprócz standardowych właściwości grup użytkowników metoda akceptuje następujące parametry.

Parameter	Type	Description
hostgroup_rights	object/array	Uprawnienia do grup hostów, które zastąpią bieżące uprawnienia przypisane do grupy użytkowników.
proxies	object/array	Proxy, które zostaną powiązane z grupą użytkowników. Proxy muszą mieć zdefiniowaną wyłącznie właściwość `proxyid`.
proxy_groups	object/array	Grupy proxy, które zostaną powiązane z grupą użytkowników. Grupy proxy muszą mieć zdefiniowaną wyłącznie właściwość `proxy_groupid`.
templategroup_rights	object/array	Uprawnienia do grup szablonów, które zastąpią bieżące uprawnienia przypisane do grupy użytkowników.
tag_filters	array	Uprawnienia oparte na tagach, które zastąpią bieżące uprawnienia przypisane do grupy użytkowników.
users	object/array	Użytkownicy, którzy zastąpią bieżących użytkowników przypisanych do grupy użytkowników. Użytkownik musi mieć zdefiniowaną wyłącznie właściwość `userid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych grup użytkowników we właściwości `usrgrpids`.

Przykłady

Włączanie grupy użytkowników i aktualizacja uprawnień

Włącz grupę użytkowników i zapewnij jej dostęp do odczytu i zapisu do grup hostów „2” i „4”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usergroup.update",
    "params": {
        "usrgrpid": "17",
        "users_status": "0",
        "hostgroup_rights": [
            {
                "id": "2",
                "permission": 3
            },
            {
                "id": "4",
                "permission": 3
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "usrgrpids": [
            "17"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uprawnienie

Źródło

CUserGroup::update() w *ui/include/classes/api/services/CUserGroup.php*.

Historia

Ta klasa jest przeznaczona do pracy z danymi historii.

Odwołania do obiektów:

- Historia typu float
- Historia typu integer
- Historia typu string
- Historia tekstowa
- Historia logów

Dostępne metody:

- history.clear - wyczyść dane historii
- history.get - pobierz dane historii
- history.push - wyślij dane historii do serwer Zabbix

Obiekt Historia

Następujące obiekty są bezpośrednio powiązane z API `history`.

Note:
Obiekty historii różnią się w zależności od typu informacji elementu. Są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pośrednictwem API.

Historia wartości zmiennoprzecinkowych

Obiekt historii wartości zmiennoprzecinkowych ma następujące właściwości.

Właściwość	Type	Opis
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	float	Otrzymana wartość.

Historia liczb całkowitych

Obiekt historii liczb całkowitych ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Czas, kiedy ta wartość została odebrana.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie odebrania wartości.
value	integer	Odebrana wartość.

Historia ciągów znaków

Obiekt historii ciągów znaków ma następujące właściwości.

Właściwość	Type	Opis
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	string	Otrzymana wartość.

Historia tekstu

Obiekt historii tekstu ma następujące właściwości.

Właściwość	Typ	Opis
id	ID	ID wpisu historii.
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	text	Otrzymana wartość.

Historia logów

Obiekt historii logów ma następujące właściwości.

Właściwość	Typ	Opis
id	ID	ID wpisu historii.
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
logeventid	integer	ID wpisu dziennika zdarzeń systemu Windows.
ns	integer	Nanosekundy w momencie otrzymania wartości.
severity	integer	Poziom wpisu dziennika zdarzeń systemu Windows.
source	string	Źródło wpisu dziennika zdarzeń systemu Windows.
timestamp	timestamp	Czas wpisu dziennika zdarzeń systemu Windows.
value	text	Otrzymana wartość.

Historia binarna

Obiekt historii binarnej ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	text	Otrzymana wartość.

Historia JSON

Obiekt historii JSON ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Czas otrzymania tej wartości.
itemid	ID	ID powiązanej pozycji.
ns	integer	Nanosekundy w momencie otrzymania wartości.
value	text	Otrzymana wartość.

history.clear

Opis

`object history.clear(array itemids)`

Ta metoda umożliwia wyczyszczenie historii pozycji.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` Identyfikatory pozycji do wyczyszczenia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory wyczyszczonych pozycji we właściwości `itemids`.

Przykłady

Wyczyść historię

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "history.clear",
"params": [
"10325",
"13205"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"wynik": {
"itemids": [
"10325",
"13205"
]
},
"id": 1
}
```

Źródło

CHistory::clear() w *ui/include/classes/api/services/CHistory.php*.

history.get

Opis

`integer/array history.get(object parameters)`

Metoda umożliwia pobranie danych historycznych zgodnie z podanymi parametrami.

> **Attention:**
> Ta metoda może zwrócić dane historyczne usuniętej encji, jeśli dane te nie zostały jeszcze usunięte przez housekeeper.

> **Note:**
> Ta metoda jest dostępna dla użytkowników każdego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
history	integer	Typy obiektów historii do zwrócenia. Możliwe wartości: 0 - liczba zmiennoprzecinkowa; 1 - znak; 2 - log; 3 - *(domyślnie)* liczba całkowita bez znaku; 4 - tekst; 5 - binarny; 6 - JSON.
hostids	ID/array	Zwraca tylko historię z podanych hostów.
itemids	ID/array	Zwraca tylko historię z podanych pozycji.
maxValueSize	integer	Maksymalna liczba bajtów do zwrócenia w właściwości `value`. Zakres możliwych wartości: 1-134217728 (1 bajt do 128 MiB). Jeśli ustawiono `null`, nie jest stosowany żaden limit. Domyślnie: 65536 (64 KiB). Zachowanie parametru: - *obsługiwane* jeśli `history` jest ustawione na ”binary” lub ”JSON”
time_from	timestamp	Zwraca tylko wartości, które zostały odebrane po podanym czasie lub dokładnie w tym czasie.
time_till	timestamp	Zwraca tylko wartości, które zostały odebrane przed podanym czasem lub dokładnie w tym czasie.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `clock`, `ns`. Nie obsługuje sortowania według `ns`, jeśli dane historii są przechowywane w ClickHouse.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Nie obsługuje wyszukiwania według `value`, jeśli `history` jest ustawione na 6 (JSON).
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych historii pozycji

Zwraca 10 najnowszych wartości otrzymanych z pozycji typu numeric(float).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "history.get",
    "params": {
        "output": "extend",
        "history": 0,
        "itemids": "23296",
        "sortfield": "clock",
        "sortorder": "DESC",
        "limit": 10
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Źródło

CHistory::get() w *ui/include/classes/api/services/CHistory.php*.

history.push

Opis

`object history.push(object/array itemHistoryData)`

Ta metoda umożliwia wysyłanie danych historii pozycji do serwer Zabbix.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji znajduje się w sekcji Role użytkowników.

Parametry

`(object/array)` Dane historii pozycji do wysłania.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
itemid	ID	ID powiązanej pozycji. Zachowanie parametru: - *wymagany*, jeśli `host` i `key` nie są ustawione
host	string	Nazwa techniczna hosta. Zachowanie parametru: - *wymagany*, jeśli `itemid` nie jest ustawione
key	string	Klucz pozycji. Zachowanie parametru: - *wymagany*, jeśli `itemid` nie jest ustawione
value	mixed	Wartość pozycji. Zachowanie parametru: - *wymagany*

Parametr	Type	Opis
clock	timestamp	Czas otrzymania wartości.
ns	integer	Nanosekundy w momencie otrzymania wartości.

Zwracane wartości

`(object)` Zwraca wynik operacji wysyłania danych.

Przykłady

Wysyłanie danych historii pozycji

Wyślij dane historii pozycji do serwera Zabbix dla pozycji „10600”, „10601” i „999999”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "response": "success",
        "data": [
            {
                "itemid": "10600"
            },
            {
                "itemid": "10600"
            },
            {
                "itemid": "10601",
                "error": "Item is disabled."
            },
            {
                "error": "No permissions to referred object or it does not exist."
            }
        ]
```

```
    },
    "id": 1
}
```

Zobacz także

- Pozycje trapper
- pozycje HTTP agent
- Host
- Pozycja

Źródło

CHistory::push() w *ui/include/classes/api/services/CHistory.php*.

Host

Ta klasa jest przeznaczona do pracy z hostami.

Odwołania do obiektów:

- Host
- Inwentarz hosta
- Tag hosta

Dostępne metody:

- host.create - utwórz nowe hosty
- host.delete - usuń hosty
- host.get - pobierz hosty
- host.massadd - dodaj powiązane obiekty do hostów
- host.massremove - usuń powiązane obiekty z hostów
- host.update - zaktualizuj hosty

Obiekt hosta

Następujące obiekty są bezpośrednio powiązane z interfejsem API hosta.

Host

Obiekt host ma następujące właściwości.

Property	Type	Description
hostid	ID	Identyfikator hosta. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
host	string	Nazwa techniczna hosta. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
description	text	Opis hosta.
flags	integer	Pochodzenie hosta. Możliwe wartości: 0 - zwykły host; 4 - host przekonwertowany z prototypu. Zachowanie właściwości: - *tylko do odczytu*

Property	Type	Description
inventory_mode	integer	Tryb uzupełniania inwentaryzacji hosta. Możliwe wartości: -1 - *(domyślnie)* wyłączone; 0 - ręczny; 1 - automatyczny.
ipmi_authtype	integer	Algorytm uwierzytelniania IPMI. Możliwe wartości: -1 - *(domyślnie)* domyślny; 0 - brak; 1 - MD2; 2 - MD5 4 - straight; 5 - OEM; 6 - RMCP+.
ipmi_password	string	Hasło IPMI.
ipmi_privilege	integer	Poziom uprawnień IPMI. Możliwe wartości: 1 - callback; 2 - *(domyślnie)* użytkownik; 3 - operator; 4 - administrator; 5 - OEM.
ipmi_username	string	Nazwa użytkownika IPMI.
maintenance_from	timestamp	Czas rozpoczęcia obowiązującego okresu konserwacji. Zachowanie właściwości: - *tylko do odczytu*
maintenance_status	integer	Obowiązujący status konserwacji. Możliwe wartości: 0 - *(domyślnie)* brak konserwacji; 1 - konserwacja obowiązuje. Zachowanie właściwości: - *tylko do odczytu*
maintenance_type	integer	Obowiązujący typ konserwacji. Możliwe wartości: 0 - *(domyślnie)* konserwacja ze zbieraniem danych; 1 - konserwacja bez zbierania danych. Zachowanie właściwości: - *tylko do odczytu*
maintenanceid	ID	Identyfikator konserwacji, która obecnie obowiązuje dla hosta. Zachowanie właściwości: - *tylko do odczytu*
name	string	Widoczna nazwa hosta. Wartość domyślna: wartość właściwości `host`.

Property	Type	Description
monitored_by	integer	Źródło używane do monitorowania hosta. Możliwe wartości: 0 - *(domyślnie)* serwer Zabbix; 1 - proxy; 2 - grupa proxy. Zachowanie właściwości: - *wymagane*, jeśli `select_server_for_monitoring` jest ustawione na „Wyłączone”.
proxyid	ID	Identyfikator proxy używanego do monitorowania hosta. Zachowanie właściwości: - *wymagane*, jeśli `monitored_by` jest ustawione na „Proxy”
proxy_groupid	ID	Identyfikator grupy proxy używanej do monitorowania hosta. Zachowanie właściwości: - *wymagane*, jeśli `monitored_by` jest ustawione na „Grupa proxy”
status	integer	Status i funkcja hosta. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
tls_connect	integer	Połączenia z hostem. Możliwe wartości: 1 - *(domyślnie)* brak szyfrowania; 2 - PSK; 4 - certyfikat.
tls_accept	integer	Połączenia z hosta. Możliwe wartości bitmapy: 1 - *(domyślnie)* brak szyfrowania; 2 - PSK; 4 - certyfikat. To pole jest maską bitową; dopuszczalna jest dowolna suma możliwych wartości bitmapy (na przykład 6 dla PSK i certyfikatu).
tls_issuer	string	Wystawca certyfikatu.
tls_subject	string	Podmiot certyfikatu.
tls_psk_identity	string	Tożsamość PSK; musi być powiązana tylko z jednym PSK (w ramach automatycznej rejestracji, hostów i proxy). Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć bez szyfrowania, aby poinformować odbiorcę, którego PSK należy użyć. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `tls_connect` jest ustawione na „PSK” lub `tls_accept` zawiera bit „PSK”
tls_psk	string	Wstępnie współdzielony klucz (PSK); musi zawierać co najmniej 32 cyfry szesnastkowe. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `tls_connect` jest ustawione na „PSK” lub `tls_accept` zawiera bit „PSK”

Property	Type	Description
active_available	integer	Status dostępności aktywnego interfejsu hosta. Możliwe wartości: 0 - status interfejsu jest nieznany; 1 - interfejs jest dostępny; 2 - interfejs jest niedostępny. Zachowanie właściwości: - *tylko do odczytu*
assigned_proxyid	ID	Identyfikator proxy przypisanego przez serwer Zabbix, jeśli host jest monitorowany przez grupę proxy. Zachowanie właściwości: - *tylko do odczytu*

Inwentaryzacja hosta

Obiekt inwentaryzacji hosta ma następujące właściwości.

Note:
Każda właściwość ma własny unikalny numer ID, który jest używany do powiązania pól inwentaryzacji hosta z pozycjami.

ID	Właściwość	Type	Opis	Maksymalna długość
4	alias	string	Alias.	128 znaków
11	asset_tag	string	Znacznik zasobu.	64 znaki
28	chassis	string	Obudowa.	64 znaki
23	contact	string	Osoba kontaktowa.	65535 znaków dla baz danych SQL
32	contract_number	string	Numer umowy.	64 znaki
47	date_hw_decomm	string	Data wycofania sprzętu z eksploatacji.	64 znaki
46	date_hw_expiry	string	Data wygaśnięcia konserwacji sprzętu.	64 znaki
45	date_hw_install	string	Data instalacji sprzętu.	64 znaki
44	date_hw_purchase	string	Data zakupu sprzętu.	64 znaki
34	deployment_status	string	Status wdrożenia.	64 znaki
14	hardware	string	Sprzęt.	255 znaków
15	hardware_full	string	Szczegółowe informacje o sprzęcie.	65535 znaków dla baz danych SQL
39	host_netmask	string	Maska podsieci hosta.	39 znaków
38	host_networks	string	Sieci hosta.	65535 znaków dla baz danych SQL
40	host_router	string	Router hosta.	39 znaków
30	hw_arch	string	Architektura sprzętu.	32 znaki
33	installer_name	string	Nazwa instalatora.	64 znaki
24	location	string	Lokalizacja.	65535 znaków dla baz danych SQL
25	location_lat	string	Szerokość geograficzna lokalizacji.	16 znaków
26	location_lon	string	Długość geograficzna lokalizacji.	16 znaków
12	macaddress_a	string	Adres MAC A.	64 znaki
13	macaddress_b	string	Adres MAC B.	64 znaki
29	model	string	Model.	64 znaki
3	name	string	Nazwa.	128 znaków
27	notes	string	Uwagi.	65535 znaków dla baz danych SQL
41	oob_ip	string	Adres IP OOB.	39 znaków
42	oob_netmask	string	Maska podsieci hosta OOB.	39 znaków
43	oob_router	string	Router OOB.	39 znaków
5	os	string	Nazwa systemu operacyjnego.	128 znaków
6	os_full	string	Pełna nazwa systemu operacyjnego.	255 znaków
7	os_short	string	Krótka nazwa systemu operacyjnego.	128 znaków
61	poc_1_cell	string	Numer telefonu komórkowego głównego punktu kontaktowego.	64 znaki
58	poc_1_email	string	Główny adres e-mail.	128 znaków
57	poc_1_name	string	Nazwa głównego punktu kontaktowego.	128 znaków
63	poc_1_notes	string	Uwagi dotyczące głównego punktu kontaktowego.	65535 znaków dla baz danych SQL

ID	Właściwość	Type	Opis	Maksymalna długość
59	poc_1_phone_a	string	Telefon A głównego punktu kontaktowego.	64 znaki
60	poc_1_phone_b	string	Telefon B głównego punktu kontaktowego.	64 znaki
62	poc_1_screen	string	Nazwa ekranowa głównego punktu kontaktowego.	64 znaki
68	poc_2_cell	string	Numer telefonu komórkowego dodatkowego punktu kontaktowego.	64 znaki
65	poc_2_email	string	Dodatkowy adres e-mail.	128 znaków
64	poc_2_name	string	Nazwa dodatkowego punktu kontaktowego.	128 znaków
70	poc_2_notes	string	Uwagi dotyczące dodatkowego punktu kontaktowego.	65535 znaków dla baz danych SQL
66	poc_2_phone_a	string	Telefon A dodatkowego punktu kontaktowego.	64 znaki
67	poc_2_phone_b	string	Telefon B dodatkowego punktu kontaktowego.	64 znaki
69	poc_2_screen	string	Nazwa ekranowa dodatkowego punktu kontaktowego.	64 znaki
8	serialno_a	string	Numer seryjny A.	64 znaki
9	serialno_b	string	Numer seryjny B.	64 znaki
48	site_address_a	string	Adres lokalizacji A.	128 znaków
49	site_address_b	string	Adres lokalizacji B.	128 znaków
50	site_address_c	string	Adres lokalizacji C.	128 znaków
51	site_city	string	Miasto lokalizacji.	128 znaków
53	site_country	string	Kraj lokalizacji.	64 znaki
56	site_notes	string	Uwagi dotyczące lokalizacji.	65535 znaków dla baz danych SQL
55	site_rack	string	Lokalizacja szafy rack w obiekcie.	128 znaków
52	site_state	string	Stan/województwo lokalizacji.	64 znaki
54	site_zip	string	Kod ZIP/pocztowy lokalizacji.	64 znaki
16	software	string	Oprogramowanie.	255 znaków
18	software_app_a	string	Aplikacja programowa A.	64 znaki
19	software_app_b	string	Aplikacja programowa B.	64 znaki
20	software_app_c	string	Aplikacja programowa C.	64 znaki
21	software_app_d	string	Aplikacja programowa D.	64 znaki
22	software_app_e	string	Aplikacja programowa E.	64 znaki
17	software_full	string	Szczegóły oprogramowania.	65535 znaków dla baz danych SQL
10	tag	string	Znacznik.	64 znaki
1	type	string	Typ.	64 znaki
2	type_full	string	Szczegóły typu.	64 znaki
35	url_a	string	URL A.	2048 znaków
36	url_b	string	URL B.	2048 znaków
37	url_c	string	URL C.	2048 znaków
31	vendor	string	Dostawca.	64 znaki

Tag hosta

Obiekt tagu hosta ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu hosta. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu hosta. Zachowanie właściwości: - *tylko do odczytu*.

Właściwość	Typ	Opis
automatic	integer	Typ tagu hosta. Możliwe wartości: 0 - *(domyślnie)* ręczny (tag utworzony przez użytkownika); 1 - automatyczny (tag utworzony przez wykrywanie niskopoziomowe) Zachowanie właściwości: - *tylko do odczytu*
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

host.create

Opis

`object host.create(object/array hosts)`

Ta metoda umożliwia tworzenie nowych hostów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Hosty do utworzenia.

Oprócz standardowych właściwości hosta, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy hostów, do których należy dodać host. Grupy hostów muszą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametru: - *wymagany*
interfaces	object/array	Interfejsy, które mają zostać utworzone dla hosta.
tags	object/array	Tagi hosta.
templates	object/array	Szablony, które mają zostać podłączone do hosta. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`.
macros	object/array	Makra użytkownika, które mają zostać utworzone dla hosta.
inventory	object	Właściwości inwentarza hosta.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych hostów we właściwości `hostids`. Kolejność zwracanych identyfikatorów odpowiada kolejności przekazanych hostów.

Przykłady

Tworzenie hosta

Utwórz host o nazwie „Linux server” z interfejsem IP i tagami, dodaj go do grupy, podłącz do niego szablon i ustaw adresy MAC w inwentarzu hosta.

Żądanie:

```
{
    "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": "latitude, longitude and altitude coordinates"
            }
        ],
        "inventory_mode": 0,
        "inventory": {
            "macaddress_a": "01234",
            "macaddress_b": "56768"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "107819"
        ]
    },
```

```
    "id": 1
}
```

Tworzenie hosta z interfejsem SNMP

Utwórz host o nazwie "SNMP host" z interfejsem SNMPv3 ze szczegółami.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10658"
        ]
    },
    "id": 1
}
```

Tworzenie hosta z szyfrowaniem PSK

Utwórz host o nazwie "PSK host" z skonfigurowanym szyfrowaniem PSK. Należy pamiętać, że host musi być wcześniej skonfigurowany do używania PSK.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

Tworzenie hosta monitorowanego przez proxy

Utwórz host, który będzie monitorowany przez proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Host monitored by proxy",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 1,
        "proxyid": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10591"
        ]
    },
    "id": 1
}
```

Tworzenie hosta monitorowanego przez grupę proxy

Utwórz host, który będzie monitorowany przez grupę proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "Host monitored by proxy group",
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "monitored_by": 2,
        "proxy_groupid": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10592"
        ]
    },
    "id": 1
}
```

Zobacz także

- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta
- Inwentaryzacja hosta
- Tag hosta
- Proxy
- Grupa proxy

Źródło

CHost::create() w *ui/include/classes/api/services/CHost.php*.

host.delete

Opis

`object host.delete(array hosts)`

Ta metoda umożliwia usuwanie hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID hostów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych hostów w właściwości `hostids`.

Przykłady

Usuwanie wielu hostów

Usuń dwa hosty.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "host.delete",
"params": [
"13",
"32"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"hostids": [
"13",
"32"
]
},
"id": 1
}
```

Źródło

CHost::delete() w *ui/include/classes/api/services/CHost.php*.

host.get

Opis

`integer/array host.get(object parameters)`

Metoda umożliwia pobieranie hostów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwracaj tylko hosty należące do podanych grup.
dserviceids	ID/array	Zwracaj tylko hosty powiązane z podanymi wykrytymi usługami.
graphids	ID/array	Zwracaj tylko hosty, które mają podane wykresy.
hostids	ID/array	Zwracaj tylko hosty o podanych identyfikatorach hostów.
httptestids	ID/array	Zwracaj tylko hosty, które mają podane testy WWW.
interfaceids	ID/array	Zwracaj tylko hosty używające podanych interfejsów.
itemids	ID/array	Zwracaj tylko hosty, które mają podane pozycje.
maintenanceids	ID/array	Zwracaj tylko hosty objęte podanymi okresami konserwacji.
monitored_hosts	flag	Zwracaj tylko monitorowane hosty.
proxyids	ID/array	Zwracaj tylko hosty monitorowane przez podane proxy.
proxy_groupids	ID/array	Zwracaj tylko hosty monitorowane przez podane grupy proxy.
templated_hosts	flag	Zwracaj zarówno hosty, jak i szablony.
templateids	ID/array	Zwracaj tylko hosty połączone z podanymi szablonami.

Parametr	Typ	Opis
triggerids	ID/array	Zwracaj tylko hosty, które mają podane wyzwalacze.
with_items	flag	Zwracaj tylko hosty, które mają pozycje. Nadpisuje parametry `with_monitored_items` i `with_simple_graph_items`.
with_item_prototypes	flag	Zwracaj tylko hosty, które mają prototypy pozycji. Nadpisuje parametr `with_simple_graph_item_prototypes`.
with_simple_graph_item_prototypes	flag	Zwracaj tylko hosty, które mają prototypy pozycji włączone do tworzenia i z numerycznym typem informacji.
with_graphs	flag	Zwracaj tylko hosty, które mają wykresy.
with_graph_prototypes	flag	Zwracaj tylko hosty, które mają prototypy wykresów.
with_httptests	flag	Zwracaj tylko hosty, które mają testy WWW. Nadpisuje parametr `with_monitored_httptests`.
with_monitored_httptests	flag	Zwracaj tylko hosty, które mają włączone testy WWW.
with_monitored_items	flag	Zwracaj tylko hosty, które mają włączone pozycje. Nadpisuje parametr `with_simple_graph_items`.
with_monitored_triggers	flag	Zwracaj tylko hosty, które mają włączone wyzwalacze. Wszystkie pozycje używane w wyzwalaczu również muszą być włączone.
with_simple_graph_items	flag	Zwracaj tylko hosty, które mają pozycje z numerycznym typem informacji.
with_triggers	flag	Zwracaj tylko hosty, które mają wyzwalacze. Nadpisuje parametr `with_monitored_triggers`.
withProblemsSuppressed	boolean	Jeśli ustawiono na `true`, zwracaj tylko hosty z wyciszonymi problemami.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
severities	integer/array	Zwracaj hosty, które mają tylko problemy o podanych poziomach ważności. Dotyczy tylko sytuacji, gdy obiektem problemu jest wyzwalacz.
tags	object/array	Zwracaj tylko hosty z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie hosty. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
inheritedTags	boolean	Zwracaj hosty, które mają podane `tags` również we wszystkich połączonych z nimi szablonach. Możliwe wartości: `true` - połączone szablony również muszą mieć podane `tags`; `false` - *(domyślnie)* tagi połączonych szablonów są ignorowane.

Parametr	Typ	Opis
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania hosta. Obiekt wykrywania hosta łączy wykryty host z prototypem hosta, z którego został wykryty. Ma następujące właściwości: `host` - `(string)` identyfikator hosta; `parent_hostid` - `(string)` identyfikator prototypu hosta, z którego host został utworzony; `status` - `(int)` status wykrywania hosta: 0 - *(domyślnie)* host jest wykryty, 1 - host nie jest już wykryty; `ts_delete` - `(timestamp)` czas, kiedy host, który nie jest już wykryty, zostanie usunięty; `ts_disable` - `(timestamp)` czas, kiedy host, który nie jest już wykryty, zostanie wyłączony; `disable_source` - `(int)` wskaźnik określający, czy host został wyłączony przez regułę LLD czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z regułą wykrywania niskiego poziomu, która utworzyła host (z prototypu hosta w monitorowaniu VMware).
selectDiscoveryRules	query	Zwracaj właściwość `discoveryRules` z regułami LLD hosta. Obsługuje `count`.
selectGraphs	query	Zwracaj właściwość `graphs` z wykresami hosta. Obsługuje `count`.
selectHostGroups	query	Zwracaj właściwość `hostgroups` z danymi grup hostów, do których należy host.
selectHttpTests	query	Zwracaj właściwość `httpTests` ze scenariuszami WWW hosta. Obsługuje `count`.
selectInterfaces	query	Zwracaj właściwość `interfaces` z interfejsami hosta. Obsługuje `count`.
selectInventory	query	Zwracaj właściwość `inventory` z danymi inwentarza hosta.
selectItems	query	Zwracaj właściwość `items` z pozycjami hosta. Obsługuje `count`.
selectMacros	query	Zwracaj właściwość `macros` z makrami hosta.
selectParentTemplates	query	Zwracaj właściwość `parentTemplates` z szablonami, z którymi host jest połączony. Oprócz pól obiektu Template zawiera ona `link_type` - `(integer)` sposób, w jaki szablon jest połączony z hostem. Możliwe wartości: 0 - *(domyślnie)* połączony ręcznie; 1 - połączony automatycznie przez LLD. Obsługuje `count`.
selectDashboards	query	Zwracaj właściwość `dashboards`. Obsługuje `count`.
selectTags	query	Zwracaj właściwość `tags` z tagami hosta.
selectInheritedTags	query	Zwracaj właściwość `inheritedTags` z tagami obecnymi we wszystkich szablonach połączonych z hostem.
selectTriggers	query	Zwracaj właściwość `triggers` z wyzwalaczami hosta. Obsługuje `count`.
selectValueMaps	query	Zwracaj właściwość `valuemaps` z mapowaniami wartości hosta.

Parametr	Typ	Opis
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: właściwości interfejsu hosta.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectParentTemplates` - wyniki będą sortowane według `host`; `selectInterfaces`; `selectItems` - sortowane według `name`; `selectDiscoveryRules` - sortowane według `name`; `selectTriggers` - sortowane według `description`; `selectGraphs` - sortowane według `name`; `selectDashboards` - sortowane według `name`.
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Obsługuje dodatkowe właściwości: właściwości interfejsu hosta.
searchInventory	object	Zwracaj hosty, których dane inwentarza pasują do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `hostid`, `host`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectDiscoveries	query	Zwracaj właściwość `discoveries` z regułami wykrywania niskiego poziomu hosta. Obsługuje `count`. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryRules`.

Parametr	Typ	Opis
selectHostDiscovery	query	Zwracaj właściwość `hostDiscovery` z danymi obiektu wykrywania hosta. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie danych według nazwy

Pobierz wszystkie dane o dwóch hostach o nazwach „Zabbix server” i „Linux server”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "filter": {
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "Serwer monitorowania Zabbix.",
            "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
}
```

Pobieranie grup hostów

Pobierz grupy hostów, do których należy host „Zabbix server”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectHostGroups": "extend",
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobieranie podłączonych szablonów

Pobierz identyfikatory i nazwy szablonów podłączonych do hosta „10084”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectParentTemplates": [
            "templateid",
            "name"
        ],
        "hostids": "10084"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "parentTemplates": [
                {
                    "name": "Linux",
                    "templateid": "10001"
                },
                {
                    "name": "Zabbix Server",
                    "templateid": "10047"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobieranie hostów według szablonu

Pobierz hosty, które mają podłączony szablon „10001” (*Linux by Zabbix agent*).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "name"],
        "templateids": "10001"
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Wyszukiwanie według danych inwentaryzacyjnych hosta

Pobierz hosty, które zawierają „Linux” w polu inwentaryzacyjnym hosta „OS”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": [
            "host"
        ],
        "selectInventory": [
            "os"
        ],
        "searchInventory": {
            "os": "Linux"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10084",
            "host": "Zabbix server",
            "inventory": {
                "os": "Linux Ubuntu"
            }
        },
        {
```

```
            "hostid": "10107",
            "host": "Linux server",
            "inventory": {
                "os": "Linux Mint"
            }
        }
    ],
    "id": 1
}
```

Wyszukiwanie według tagów hosta

Pobierz hosty, które mają tag ”host-name” równy ”linux-server”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "linux-server",
                "operator": 1
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10085",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "linux-server",
                    "automatic": "0"
                },
                {
                    "tag": "os",
                    "value": "rhel-7",
                    "automatic": "0"
                }
            ]
        }
    ],
    "id": 1
}
```

Pobierz hosty, które mają te tagi nie tylko na poziomie hosta, ale także w swoich podłączonych nadrzędnych szablonach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
```

```
        "tags": [
            {
                "tag": "os",
                "value": "rhel-7",
                "operator": 1
            }
        ],
        "inheritedTags": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10623",
            "name": "PC room 1"
        },
        {
            "hostid": "10601",
            "name": "Office"
        }
    ],
    "id": 1
}
```

Pobierz hosty z wyłączeniem tych, które mają tag ”component”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "component",
                "value": "",
                "operator": 5
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Wyszukiwanie hosta za pomocą tagów i tagów szablonu

Pobierz host z tagami oraz wszystkimi tagami, które są powiązane z nadrzędnymi szablonami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "hostids": 10502,
        "selectTags": ["tag", "value"],
        "selectInheritedTags": ["tag", "value"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Wyszukiwanie hostów według ważności problemu

Pobierz hosty, które mają problemy o ważności „Disaster”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "severities": 5
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "10160",
            "name": "Zabbix server"
        }
    ],
    "id": 1
}
```

Pobierz hosty, które mają problemy o ważności „Average” i „High”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["name"],
        "severities": [3, 4]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostid": "20170",
            "name": "Database"
        },
```

```
        {
            "hostid": "20183",
            "name": "workstation"
        }
    ],
    "id": 1
}
```

Pobieranie pozycji hosta

Pobierz pozycje dla hosta „Zabbix server”. Żądanie jest ograniczone do właściwości pozycji `itemid`, `name` i `status`. Ze względu na duży rozmiar odpowiedzi w przykładzie pokazano tylko podzbiór pozycji.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid"],
        "selectItems": ["itemid","name","status"],
        "filter": {
            "host": [
                "Zabbix server"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "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
}
```


Zobacz także

- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta
- Proxy
- Grupa proxy

Źródło

CHost::get() w *ui/include/classes/api/services/CHost.php*.

host.massadd

Opis

`object host.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wszystkich podanych hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory hostów do zaktualizowania oraz obiekty do dodania do wszystkich hostów.

Metoda akceptuje następujące parametry.

Parametr	Type	Opis
hosts	object/array	Hosty do zaktualizowania. Hosty muszą mieć zdefiniowaną wyłącznie właściwość `hostid`. Zachowanie parametrów: - *wymagane*
groups	object/array	Grupy hostów do dodania do podanych hostów. Grupy hostów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`.
interfaces	object/array	Interfejsy hostów do utworzenia dla podanych hostów.
macros	object/array	Makra użytkownika do utworzenia dla podanych hostów.
templates	object/array	Szablony do podłączenia do podanych hostów. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych hostów we właściwości `hostids`.

Przykłady

Dodawanie makr

Dodaj dwa nowe makra do dwóch hostów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.massadd",
    "params": {
        "hosts": [
            {
                "hostid": "10160"
            },
            {
                "hostid": "10167"
            }
        ],
        "macros": [
            {
                "macro": "{$TEST1}",
                "value": "MACROTEST1"
            },
            {
                "macro": "{$TEST2}",
                "value": "MACROTEST2",
```

```
                "description": "Opis testowy"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10160",
            "10167"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.update
- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta

Źródło

CHost::massAdd() w *ui/include/classes/api/services/CHost.php*.

host.massremove

Opis

`object host.massremove(object parameters)`

Ta metoda umożliwia usuwanie powiązanych obiektów z wielu hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory hostów do zaktualizowania oraz obiekty, które należy usunąć.

Parametr	Typ	Opis
hostids	ID/array	Identyfikatory hostów, które mają zostać zaktualizowane. Zachowanie parametru: - *wymagany*
groupids	ID/array	Identyfikatory grup hostów, z których należy usunąć podane hosty.
interfaces	object/array	Interfejsy hosta do usunięcia z podanych hostów. Obiekt interfejsu hosta musi mieć zdefiniowane tylko właściwości `ip`, `dns` i `port`.
macros	string/array	Makra użytkownika do usunięcia z podanych hostów.
templateids	ID/array	Identyfikatory szablonów, które należy odłączyć od podanych hostów.
templateids_clear	ID/array	Identyfikatory szablonów, które należy odłączyć i wyczyścić z podanych hostów.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych hostów we właściwości hostids.

Przykłady

Odłączanie szablonów

Odłącz szablon od dwóch hostów i usuń wszystkie encje pochodzące z szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.massremove",
    "params": {
        "hostids": ["69665", "69666"],
        "templateids_clear": "325"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "69665",
            "69666"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.update
- Makro użytkownika
- Interfejs hosta

Źródło

CHost::massRemove() w *ui/include/classes/api/services/CHost.php*.

host.update

Opis

`object host.update(object/array hosts)`

Ta metoda umożliwia aktualizację istniejących hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(object/array) Właściwości hosta do zaktualizowania.

Właściwość hostid musi być zdefiniowana dla każdego hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Należy jednak pamiętać, że aktualizacja technicznej nazwy hosta spowoduje również aktualizację widocznej nazwy hosta (jeśli nie została podana osobno) na wartość technicznej nazwy hosta.

Oprócz standardowych właściwości hosta, metoda akceptuje następujące parametry.

Parameter	Type	Description
groups	object/array	Grupy hostów zastępujące bieżące grupy hostów, do których należy host. Wszystkie grupy hostów, które nie zostały wymienione w żądaniu, zostaną odłączone. Grupy hostów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`.
interfaces	object/array	Interfejsy hosta zastępujące bieżące interfejsy hosta. Wszystkie interfejsy, które nie zostały wymienione w żądaniu, zostaną usunięte.
tags	object/array	Tagi hosta zastępujące bieżące tagi hosta. Wszystkie tagi, które nie zostały wymienione w żądaniu, zostaną usunięte.
inventory	object	Właściwości inwentarza hosta.
macros	object/array	Makra użytkownika zastępujące bieżące makra użytkownika. Wszystkie makra, które nie zostały wymienione w żądaniu, zostaną usunięte.
templates	object/array	Szablony zastępujące aktualnie podłączone szablony. Wszystkie szablony, które nie zostały wymienione w żądaniu, zostaną jedynie odłączone. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.
templates_clear	object/array	Szablony do odłączenia i wyczyszczenia z hosta. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych hostów we właściwości `hostids`.

Przykłady

Włączanie monitorowania hosta

Włącz monitorowanie hosta, czyli ustaw jego status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Włączanie wielu hostów

Włącz monitorowanie dwóch hostów, tj. ustaw ich status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": [
        {
            "hostid": "10127",
            "status": 0
        },
        {
            "hostid": "10128",
            "status": 0
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10127",
            "10128"
        ]
    },
    "id": 1
}
```

Odłączanie szablonów

Odłącz i wyczyść dwa szablony z hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "templates_clear": [
            {
                "templateid": "10124"
            },
            {
                "templateid": "10125"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Aktualizowanie makr hosta

Zastąp wszystkie makra hosta dwoma nowymi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10126",
        "macros": [
            {
                "macro": "{$PASS}",
                "value": "password"
            },
            {
                "macro": "{$DISC}",
                "value": "sda",
                "description": "Updated description"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10126"
        ]
    },
    "id": 1
}
```

Aktualizowanie inwentarza hosta

Zmień tryb inwentarza i dodaj lokalizację

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "inventory_mode": 0,
        "inventory": {
            "location": "Latvia, Riga"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Aktualizowanie tagów hosta

Zastąp wszystkie tagi hosta nowym tagiem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "tags": {
            "tag": "os",
            "value": "rhel-7"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Aktualizowanie makr wykrytego hosta

Przekształć makro utworzone przez regułę wykrywania z trybu „automatic” na „manual” i zmień jego wartość na „new-value”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.update",
    "params": {
        "hostid": "10387",
        "macros": {
            "hostmacroid": "5541",
            "value": "new-value",
            "automatic": "0"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10387"
        ]
    },
    "id": 1
}
```

Aktualizacja szyfrowania hosta

Zaktualizuj host ”10590”, aby używał szyfrowania PSK tylko dla połączeń z hosta do serwera Zabbix, i zmień identyfikator PSK oraz klucz PSK. Należy pamiętać, że host musi być wcześniej skonfigurowany do używania PSK.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10590"
        ]
    },
    "id": 1
}
```

Zobacz także

- host.massadd
- host.massremove
- Grupa hostów
- Szablon
- Makro użytkownika
- Interfejs hosta
- Inwentarz hosta
- Tag hosta
- Proxy
- Grupa proxy

Źródło

CHost::update() w *ui/include/classes/api/services/CHost.php*.

Informacje o serwerze

Ta klasa służy do pobierania informacji o serwerze Zabbix.

Dostępne metody:

- serverinfo.get - pobieranie czasu ostatniej aktualizacji z serwera Zabbix

serverinfo.get

Ta metoda umożliwia pobranie czasu ostatniej aktualizacji z danego serwera Zabbix.

Metoda jest dostępna dla użytkowników z rolami *User*, *Admin*, *Super admin* oraz dla użytkowników nieuwierzytelnionych.

Sygnatura metody: `object serverinfo.get(object parameters)`

Źródło metody: `CServerInfo::get()` w pliku `ui/include/classes/api/services/CServerInfo.php`

Parametry

Parameter	Type	Opis
serverid	ID	ID serwera Zabbix. Zachowanie parametru: - Wymagany

Parameter	Type	Opis
output	query	Ten parametr został opisany w komentarzu referencyjnym.

Wartości zwracane

Właściwość	Typ	Opis
serverid	ID	Identyfikator serwera Zabbix.
lastaccess	timestamp	Czas ostatniej aktualizacji z serwera Zabbix. W przypadku klastra wysokiej dostępności (HA) wyświetlana jest wartość `lastaccess` aktywnego węzła.

Przykłady

Pobieranie informacji o serwerze

Pobiera informacje o określonym serwerze Zabbix.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "serverinfo.get",
    "params": {
        "serverid": "019fb1f9-22bd-74ea-bc03-51efb8b76aba",
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serverid": "019fb1f9-22bd-74ea-bc03-51efb8b76aba",
        "lastaccess": 1785337245
    },
    "id": 1
}
```

Interfejs hosta

Ta klasa zapewnia dostęp do interfejsów hosta.

Odwołania do obiektów:

- Interfejs hosta
  - Szczegóły

Dostępna metoda:

- hostinterface.get - pobieranie interfejsów hosta

Interfejsy hosta są tworzone, aktualizowane i usuwane za pomocą metod hosta.

Obiekt interfejsu hosta

Następujące obiekty są bezpośrednio powiązane z API `hostinterface`.

Interfejs hosta

Obiekt interfejsu hosta ma następujące właściwości.

Property	Type	Description
interfaceid	ID	Identyfikator interfejsu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w przypadku operacji aktualizacji
available	integer	Dostępność interfejsu hosta. Możliwe wartości: 0 - *(domyślnie)* nieznana; 1 - dostępny; 2 - niedostępny. Zachowanie właściwości: - *tylko do odczytu*
hostid	ID	Identyfikator hosta, do którego należy interfejs. Zachowanie właściwości: - *stałe* - *wymagane* w przypadku operacji tworzenia
type	integer	Typ interfejsu. Możliwe wartości: 1 - agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia
ip	string	Adres IP używany przez interfejs. Może być pusty, jeśli połączenie jest nawiązywane za pośrednictwem DNS. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia
dns	string	Nazwa DNS używana przez interfejs. Może być pusta, jeśli połączenie jest nawiązywane za pośrednictwem adresu IP. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia
port	string	Numer portu używany przez interfejs. Może zawierać makra użytkownika. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia
useip	integer	Określa, czy połączenie powinno być nawiązywane za pośrednictwem adresu IP. Możliwe wartości: 0 - połączenie przy użyciu nazwy DNS hosta; 1 - połączenie przy użyciu adresu IP hosta. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia

Property	Type	Description
main	integer	Określa, czy interfejs jest używany jako domyślny na hoście. Tylko jeden interfejs danego typu może być ustawiony jako domyślny na hoście. Możliwe wartości: 0 - niedomyślny; 1 - domyślny. Zachowanie właściwości: - *wymagane* w przypadku operacji tworzenia
details	object	Dodatkowy obiekt details interfejsu. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SNMP”
disable_until	timestamp	Następny czas odpytywania niedostępnego interfejsu hosta. Zachowanie właściwości: - *tylko do odczytu*
error	string	Tekst błędu, jeśli interfejs hosta jest niedostępny. Zachowanie właściwości: - *tylko do odczytu*
errors_from	timestamp	Czas, w którym interfejs hosta stał się niedostępny. Zachowanie właściwości: - *tylko do odczytu*

Szczegóły

Obiekt szczegółów ma następujące właściwości.

Właściwość	Typ	Opis
version	integer	Wersja interfejsu SNMP. Możliwe wartości: 1 - SNMPv1; 2 - SNMPv2c; 3 - SNMPv3. Zachowanie właściwości: - *wymagane*
bulk	integer	Czy używać zbiorczych żądań SNMP. Możliwe wartości: 0 - nie używaj zbiorczych żądań; 1 - (domyślnie) - używaj zbiorczych żądań.
community	string	Społeczność SNMP. Używana tylko przez interfejsy SNMPv1 i SNMPv2. Zachowanie właściwości: - *wymagane* jeśli `version` jest ustawione na ”SNMPv1” lub ”SNMPv2c”
max_repetitions	integer	Maksymalna wartość powtórzeń dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs). Używane tylko dla pozycji `discovery[]` i `walk[]` w SNMPv2 i v3. Domyślnie: 10.
securityname	string	Nazwa bezpieczeństwa SNMPv3. Używana tylko przez interfejsy SNMPv3.

Właściwość	Typ	Opis
securitylevel	integer	Poziom bezpieczeństwa SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - (domyślnie) - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv.
authpassphrase	string	Hasło uwierzytelniania SNMPv3. Używane tylko przez interfejsy SNMPv3.
privpassphrase	string	Hasło prywatności SNMPv3. Używane tylko przez interfejsy SNMPv3.
authprotocol	integer	Protokół uwierzytelniania SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - (domyślnie) - MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512.
privprotocol	integer	Protokół prywatności SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - (domyślnie) - DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C.
contextname	string	Nazwa kontekstu SNMPv3. Używana tylko przez interfejsy SNMPv3.

hostinterface.get

Opis

`integer/array hostinterface.get(object parameters)`

Metoda umożliwia pobieranie interfejsów hosta zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
hostids	ID/array	Zwraca tylko interfejsy hosta używane przez podane hosty.
interfaceids	ID/array	Zwraca tylko interfejsy hosta o podanych identyfikatorach.
itemids	ID/array	Zwraca tylko interfejsy hosta używane przez podane pozycje.
triggerids	ID/array	Zwraca tylko interfejsy hosta używane przez pozycje w podanych wyzwalaczach.
selectItems	query	Zwraca właściwość `items` z pozycjami, które używają interfejsu. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów używających interfejsu.

Parametr	Typ	Opis
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectItems`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `interfaceid`, `dns`, `ip`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie interfejsów hosta

Pobierz wszystkie dane o interfejsach używanych przez host „30057”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostinterface.get",
    "params": {
        "output": "extend",
        "hostids": "30057"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Host
- Item

Źródło

CHostInterface::get() w *ui/include/classes/api/services/CHostInterface.php*.

Katalog użytkowników

Ta klasa jest przeznaczona do pracy z katalogami użytkowników.

Odwołania do obiektów:

- Katalog użytkowników
  - Mapowania typów mediów
  - Mapowania grup provisioningu

Dostępne metody:

- userdirectory.create - utwórz nowy katalog użytkowników
- userdirectory.delete - usuń katalog użytkowników
- userdirectory.get - pobierz katalog użytkowników
- userdirectory.update - zaktualizuj katalog użytkowników
- userdirectory.test - przetestuj połączenie z katalogiem użytkowników

Obiekt katalogu użytkownika

Następujące obiekty są bezpośrednio powiązane z API `userdirectory`.

Katalog użytkowników

Obiekt katalogu użytkowników ma następujące właściwości.

Właściwość	Typ	Opis
userdirectoryid	ID	ID katalogu użytkowników. Jeśli katalog użytkowników zostanie usunięty, wartość właściwości `userdirectoryid` obiektu User jest ustawiana na „0” dla wszystkich użytkowników powiązanych z usuniętym katalogiem użytkowników. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* w operacjach aktualizacji
idp_type	integer	Typ protokołu uwierzytelniania używanego przez dostawcę tożsamości dla katalogu użytkowników. Należy pamiętać, że może istnieć tylko jeden katalog użytkowników typu SAML. Możliwe wartości: 1 - katalog użytkowników typu LDAP; 2 - katalog użytkowników typu SAML. Zachowanie właściwości: - *wymagana* w operacjach tworzenia
group_name	string	Atrybut katalogu użytkowników LDAP/SAML zawierający nazwę grupy używaną do mapowania grup między katalogiem użytkowników LDAP/SAML a Zabbixem. Przykład: *cn* Zachowanie właściwości: - *wymagana*, jeśli `provision_status` ma wartość „Enabled”, a `saml_jit_status` obiektu Authentication ma wartość „Enabled for configured SAML IdPs”
user_username	string	Atrybut katalogu użytkowników LDAP/SAML (również atrybut SCIM, jeśli `scim_status` ma wartość „SCIM provisioning is enabled”) zawierający nazwę użytkownika, która jest używana jako wartość właściwości `name` obiektu User podczas aprowizowania użytkownika. Przykłady: *cn*, *commonName*, *displayName*, *name*
user_lastname	string	Atrybut katalogu użytkowników LDAP/SAML (również atrybut SCIM, jeśli `scim_status` ma wartość „SCIM provisioning is enabled”) zawierający nazwisko użytkownika, które jest używane jako wartość właściwości `surname` obiektu User podczas aprowizowania użytkownika. Przykłady: *sn*, *surname*, *lastName*
provision_status	integer	Stan aprowizowania katalogu użytkowników. Możliwe wartości: 0 - *(domyślnie)* Wyłączone (aprowizowanie użytkowników utworzonych przez ten katalog użytkowników jest wyłączone); 1 - Włączone (aprowizowanie użytkowników utworzonych przez ten katalog użytkowników jest włączone; dodatkowo stan aprowizowania LDAP lub SAML (`ldap_jit_status` lub `saml_jit_status` obiektu Authentication) musi być włączony).
provision_groups	array	Tablica obiektów mapowań grup aprowizowania służących do mapowania wzorca grupy użytkowników LDAP/SAML na grupę użytkowników i rolę użytkownika w Zabbixie. Zachowanie właściwości: - *wymagana*, jeśli `provision_status` ma wartość „Enabled”
provision_media	array	Tablica obiektów mapowań typów mediów służących do mapowania atrybutów mediów użytkownika LDAP/SAML (np. adresu e-mail) na media użytkownika Zabbixa używane do wysyłania powiadomień.

Właściwość	Typ	Opis
Właściwości specyficzne dla LDAP:		
name	string	Unikalna nazwa katalogu użytkowników. Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
host	string	Nazwa hosta, adres IP lub URI serwera LDAP. URI musi zawierać schemat (`ldap://` lub `ldaps://`), host oraz opcjonalnie port. Przykłady: *host.example.com* *127.0.0.1* *ldap://ldap.example.com:389* Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
port	integer	Port serwera LDAP. Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
base_dn	string	Ścieżka bazowa katalogu użytkowników LDAP do kont użytkowników. Przykłady: *ou=Users,dc=example,dc=org* *ou=Users,ou=system* (dla OpenLDAP) *DC=company,DC=com* (dla Microsoft Active Directory) *uid=%{user},dc=example,dc=com* (dla bezpośredniego wiązania użytkownika; symbol zastępczy *„%{user}”* jest wymagany) Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
search_attribute	string	Atrybut katalogu użytkowników LDAP, według którego identyfikowane jest konto użytkownika na podstawie informacji podanych w żądaniu logowania. Przykłady: *uid* (dla OpenLDAP) *sAMAccountName* (dla Microsoft Active Directory) Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
bind_dn	string	Konto serwera LDAP używane do wiązania i wyszukiwania na serwerze LDAP. W przypadku bezpośredniego wiązania użytkownika i wiązania anonimowego `bind_dn` musi być puste. Przykłady: *uid=ldap_search,ou=system* (dla OpenLDAP) *CN=ldap_search,OU=user_group,DC=company,DC=com* (dla Microsoft Active Directory) *CN=Admin,OU=Users,OU=Zabbix,DC=zbx,DC=local* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”

Właściwość	Typ	Opis
bind_password	string	Hasło LDAP konta używanego do wiązania i wyszukiwania na serwerze LDAP. W przypadku bezpośredniego wiązania użytkownika i wiązania anonimowego `bind_password` musi być puste. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
description	string	Opis katalogu użytkowników. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
group_basedn	string	Ścieżka bazowa katalogu użytkowników LDAP do grup; używana do konfigurowania sprawdzania członkostwa użytkownika w katalogu użytkowników LDAP. Ignorowana podczas aprowizowania użytkownika, jeśli ustawiono `group_membership`. Przykład: *ou=Groups,dc=example,dc=com* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
group_filter	string	Ciąg filtru służący do pobierania grup katalogu użytkowników LDAP, których członkiem jest użytkownik; używany do konfigurowania sprawdzania członkostwa użytkownika w katalogu użytkowników LDAP. Ignorowany podczas aprowizowania użytkownika, jeśli ustawiono `group_membership`. Obsługiwane symbole zastępcze `group_filter`: *%{attr}* - atrybut wyszukiwania (zastępowany wartością właściwości `search_attribute`); *%{groupattr}* - atrybut grupy (zastępowany wartością właściwości `group_member`); *%{host}* - nazwa hosta, adres IP lub URI serwera LDAP (zastępowany wartością właściwości `host`); *%{user}* - nazwa użytkownika Zabbixa. Wartość domyślna: *(%{groupattr}=%{user})* Przykłady: - *(member=uid=%{ref},ou=Users,dc=example,dc=com)* dopasuje „User1”, jeśli obiekt grupy LDAP zawiera atrybut „*member*” z wartością „*uid=User1,ou=Users,dc=example,dc=com*”, i zwróci grupę, której członkiem jest „User1”; - *(%{groupattr}=cn=%{ref},ou=Users,ou=Zabbix,DC=example,DC=com)* dopasuje „User1”, jeśli obiekt grupy LDAP zawiera atrybut określony we właściwości `group_member` z wartością „*cn=User1,ou=Users,ou=Zabbix,DC=example,DC=com*”, i zwróci grupę, której członkiem jest „User1”. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”

Właściwość	Typ	Opis
group_member	string	Atrybut katalogu użytkowników LDAP zawierający informacje o członkach grupy; używany do konfigurowania sprawdzania członkostwa użytkownika w katalogu użytkowników LDAP. Ignorowany podczas aprowizowania użytkownika, jeśli ustawiono `group_membership`. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
group_membership	string	Atrybut katalogu użytkowników LDAP zawierający informacje o grupach, do których należy użytkownik. Przykład: *memberOf* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
search_filter	string	Niestandardowy ciąg filtru używany do lokalizowania i uwierzytelniania użytkownika w katalogu użytkowników LDAP na podstawie informacji podanych w żądaniu logowania. Obsługiwane symbole zastępcze `search_filter`: *%{attr}* - nazwa atrybutu wyszukiwania (np. *uid*, *sAMAccountName*); *%{user}* - nazwa użytkownika Zabbixa. Wartość domyślna: *(%{attr}=%{user})* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
start_tls	integer	Opcja konfiguracji serwera LDAP umożliwiająca zabezpieczenie komunikacji z serwerem LDAP za pomocą protokołu Transport Layer Security (TLS). Należy pamiętać, że dla hostów używających protokołu `ldaps://` parametr `start_tls` musi mieć wartość „Disabled”. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
user_ref_attr	string	Atrybut katalogu użytkowników LDAP używany do odwoływania się do obiektu użytkownika. Wartość `user_ref_attr` służy do pobierania wartości z określonego atrybutu w katalogu użytkowników i umieszczania ich zamiast symbolu zastępczego *%{ref}* w ciągu `group_filter`. Przykłady: *cn*, *uid*, *member*, *uniqueMember* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type LDAP”
Właściwości specyficzne dla SAML:		

Właściwość	Typ	Opis
idp_entityid	string	URI identyfikujący dostawcę tożsamości, używany do komunikacji z dostawcą tożsamości w komunikatach SAML. Przykład: *https://idp.example.com/idp* Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type SAML”
sp_entityid	string	URL lub dowolny ciąg identyfikujący dostawcę usług dostawcy tożsamości. Przykłady: *https://idp.example.com/sp* *zabbix* Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type SAML”
username_attribute	string	Atrybut katalogu użytkowników SAML (również atrybut SCIM, jeśli `scim_status` ma wartość „SCIM provisioning is enabled”) zawierający nazwę użytkownika, która podczas uwierzytelniania jest porównywana z wartością właściwości `username` obiektu User. Przykłady: *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* Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type SAML”
sso_url	string	URL usługi jednokrotnego logowania SAML dostawcy tożsamości, do której Zabbix będzie wysyłać żądania uwierzytelniania SAML. Przykład: *http://idp.example.com/idp/sso/saml* Zachowanie właściwości: - *wymagana*, jeśli `idp_type` ma wartość „User directory of type SAML”
slo_url	string	URL usługi jednokrotnego wylogowania SAML dostawcy tożsamości, do której Zabbix będzie wysyłać żądania wylogowania SAML. Przykład: *https://idp.example.com/idp/slo/saml* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
encrypt_nameid	integer	Określa, czy identyfikator nazwy SAML powinien być szyfrowany. Możliwe wartości: 0 - *(domyślnie)* Nie szyfruj identyfikatora nazwy; 1 - Szyfruj identyfikator nazwy. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
encrypt_assertions	integer	Określa, czy asercje SAML powinny być szyfrowane. Możliwe wartości: 0 - *(domyślnie)* Nie szyfruj asercji; 1 - Szyfruj asercje. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”

Właściwość	Typ	Opis
nameid_format	string	Format identyfikatora nazwy dostawcy usług dostawcy tożsamości SAML. Przykłady: *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* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
scim_status	integer	Określa, czy aprowizowanie SCIM dla SAML jest włączone, czy wyłączone. Możliwe wartości: 0 - *(domyślnie)* Aprowizowanie SCIM jest wyłączone; 1 - Aprowizowanie SCIM jest włączone. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
sign_assertions	integer	Określa, czy asercje SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - Nie podpisuj asercji; 1 - *(domyślnie)* Podpisuj asercje. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
sign_authn_requests	integer	Określa, czy żądania AuthN SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - Nie podpisuj żądań AuthN; 1 - *(domyślnie)* Podpisuj żądania AuthN. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
sign_messages	integer	Określa, czy komunikaty SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - Nie podpisuj komunikatów; 1 - *(domyślnie)* Podpisuj komunikaty. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
sign_logout_requests	integer	Określa, czy żądania wylogowania SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - Nie podpisuj żądań wylogowania; 1 - *(domyślnie)* Podpisuj żądania wylogowania. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”

Właściwość	Typ	Opis
sign_logout_responses	integer	Określa, czy odpowiedzi wylogowania SAML powinny być podpisywane podpisem SAML. Możliwe wartości: 0 - Nie podpisuj odpowiedzi wylogowania; 1 - *(domyślnie)* Podpisuj odpowiedzi wylogowania. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”
idp_certificate	string	Zawartość certyfikatu dostawcy usług (SP) używanego do konfiguracji usługi jednokrotnego logowania (SSO) SAML. Przykłady: *-----BEGIN CERTIFICATE----- ...more encoded data... -----END CERTIFICATE-----* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”, a `$SSO['CERT_STORAGE']` ma wartość `database` w pliku konfiguracji frontendu - *tylko do zapisu*
sp_private_key	string	Zawartość klucza prywatnego dostawcy usług (SP) używanego do konfiguracji usługi jednokrotnego logowania (SSO) SAML. Zapewnia bezpieczne uwierzytelnianie i wymianę danych z dostawcą tożsamości (IdP). Przykłady: *-----BEGIN CERTIFICATE----- ...more encoded data... -----END CERTIFICATE-----* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”, a `$SSO['CERT_STORAGE']` ma wartość `database` w pliku konfiguracji frontendu - *tylko do zapisu*
sp_certificate	string	Zawartość certyfikatu dostawcy usług (SP) używanego do konfiguracji usługi jednokrotnego logowania (SSO) SAML. Przykłady: *-----BEGIN CERTIFICATE----- ...more encoded data... -----END CERTIFICATE-----* Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”, a `$SSO['CERT_STORAGE']` ma wartość `database` w pliku konfiguracji frontendu - *tylko do zapisu*
idp_certificate_hash	string	Skrót md5 wartości idp_certificate. Dla pustego idp_certificate zwracany jest pusty ciąg. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”, a `$SSO['CERT_STORAGE']` ma wartość `database` w pliku konfiguracji frontendu - *tylko do odczytu*

Właściwość	Typ	Opis
sp_private_key_hash	string	Skrót md5 wartości sp_private_key. Dla pustego sp_private_key zwracany jest pusty ciąg. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”, a `$SSO['CERT_STORAGE']` ma wartość `database` w pliku konfiguracji frontendu - *tylko do odczytu*
sp_certificate_hash	string	Skrót md5 wartości sp_certificate. Dla pustego sp_certificate zwracany jest pusty ciąg. Zachowanie właściwości: - *obsługiwana*, jeśli `idp_type` ma wartość „User directory of type SAML”, a `$SSO['CERT_STORAGE']` ma wartość `database` w pliku konfiguracji frontendu - *tylko do odczytu*

Mapowania typów mediów

Obiekt mapowań typów mediów ma następujące właściwości.

Właściwość	Typ	Opis
userdirectory_mediaid	ID	ID mapowania typu mediów. Zachowanie właściwości: - *tylko do odczytu*
name	string	Widoczna nazwa na liście mapowań typów mediów. Zachowanie właściwości: - *wymagane*
mediatypeid	ID	ID typu mediów, który ma zostać utworzony; używane jako wartość właściwości `mediatypeid` obiektu Media. Zachowanie właściwości: - *wymagane*
attribute	string	Atrybut katalogu użytkowników LDAP/SAML (również atrybut SCIM, jeśli `scim_status` jest ustawione na „Udostępnianie SCIM jest włączone”), który zawiera media użytkownika (np. *user@example.com*), używane jako wartość właściwości `sendto` obiektu Media. Jeśli występuje w danych otrzymanych od dostawcy tożsamości LDAP/SAML, a wartość nie jest pusta, spowoduje to utworzenie mediów dla udostępnionego użytkownika. Zachowanie właściwości: - *wymagane*
active	integer	Wartość właściwości `active` mediów użytkownika podczas tworzenia mediów dla udostępnionego użytkownika. Możliwe wartości: 0 - *(domyślnie)* włączone; 1 - wyłączone.
severity	integer	Wartość właściwości `severity` mediów użytkownika podczas tworzenia mediów dla udostępnionego użytkownika. Domyślnie: 63.
period	string	Wartość właściwości `period` mediów użytkownika podczas tworzenia mediów dla udostępnionego użytkownika. Domyślnie: 1-7,00:00-24:00.

Mapowania grup provisioningu

Mapowania grup provisioningu mają następujące właściwości.

Właściwość	Typ	Opis
name	string	Pełna nazwa grupy (np. *Zabbix administrators*) w katalogu użytkowników LDAP/SAML (również SCIM, jeśli `scim_status` jest ustawione na „SCIM provisioning is enabled”). Obsługuje znak wieloznaczny „*”. Unikalna wśród wszystkich mapowań grup provisioningu. Zachowanie właściwości: - *wymagane*
roleid	ID	ID roli użytkownika, która ma zostać przypisana użytkownikowi. Jeśli dopasowanych zostanie wiele mapowań grup provisioningu, użytkownikowi zostanie przypisana rola o najwyższym typie użytkownika (*User*, *Admin* lub *Super admin*). Jeśli istnieje wiele ról z tym samym typem użytkownika, użytkownikowi zostanie przypisana pierwsza rola (posortowana alfabetycznie). Zachowanie właściwości: - *wymagane*
user_groups	array	Tablica obiektów ID grup użytkowników Zabbix. Każdy obiekt ma następujące właściwości: `usrgrpid` - `(ID)` ID grupy użytkowników Zabbix, która ma zostać przypisana użytkownikowi. Jeśli dopasowanych zostanie wiele mapowań grup provisioningu, użytkownikowi zostaną przypisane grupy użytkowników Zabbix ze wszystkich dopasowanych mapowań. Zachowanie właściwości: - *wymagane*

userdirectory.create

Opis

`object userdirectory.create(object/array userDirectory)`

Ta metoda umożliwia tworzenie nowych katalogów użytkowników.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*.

Parametry

`(object/array)` Katalogi użytkowników do utworzenia.

Metoda akceptuje katalogi użytkowników ze standardowymi właściwościami katalogu użytkownika.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych katalogów użytkowników we właściwości `userdirectoryids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych katalogów użytkowników.

Przykłady

Tworzenie katalogu użytkowników

Utwórz katalog użytkowników, aby uwierzytelniać użytkowników za pomocą StartTLS przez LDAP. Pamiętaj, że aby uwierzytelniać użytkowników przez LDAP, musi być włączone uwierzytelnianie LDAP.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Tworzenie katalogu użytkowników (z włączonym udostępnianiem JIT)

Utwórz katalog użytkowników, aby uwierzytelniać użytkowników przez LDAP (z włączonym udostępnianiem JIT). Pamiętaj, że aby uwierzytelniać użytkowników przez LDAP, należy włączyć uwierzytelnianie LDAP.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CUserDirectory::create() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.delete

Opis

`object userdirectory.delete(array userDirectoryIds)`

Ta metoda umożliwia usuwanie katalogów użytkowników. Katalogu użytkowników nie można usunąć, gdy jest on bezpośrednio używany przez co najmniej jedną grupę użytkowników.
 Domyślnego katalogu użytkowników LDAP nie można usunąć, gdy `authentication.ldap_configured` ma ustawioną wartość 1 lub gdy pozostałoby mniej katalogów użytkowników.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*.

Parametry

`(tablica)` Identyfikatory katalogów użytkownika do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych katalogów użytkownika w ramach właściwości `userdirectoryids`.

Przykłady

Usuwanie wielu katalogów użytkowników

Usuń dwa katalogi użytkowników.

Żądanie:

```
{
"jsonrpc": "2.0",
```

```
"method": "userdirectory.delete",
"params": [
"2",
"12"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"userdirectoryids": [
"2",
"12"
]
},
"id": 1
}
```

Źródło

CUserDirectory::delete() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.get

Opis

`integer/array userdirectory.get(object parameters)`

Ta metoda umożliwia pobieranie katalogów użytkowników zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Super admin*.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
userdirectoryids	ID/array	Zwracaj tylko katalogi użytkowników o podanych identyfikatorach.
selectUsrgrps	query	Zwracaj właściwość `usrgrps` z grupami użytkowników powiązanymi z katalogiem użytkowników. Obsługuje `count`.
selectProvisionMedia	query	Zwracaj właściwość `provision_media` z mapowaniami typów mediów powiązanymi z katalogiem użytkowników.
selectProvisionGroups	query	Zwracaj właściwość `provision_groups` z mapowaniami grup provisioningu powiązanymi z katalogiem użytkowników.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `name`.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości. Obsługiwane właściwości: `userdirectoryid`, `idp_type`, `provision_status`.

Parametr	Typ	Opis
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są ciągami znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługiwane właściwości: `name`, `description`. Katalog użytkowników typu SAML będzie miał pustą wartość zarówno dla właściwości `name`, jak i `description`. Obie właściwości można zmienić za pomocą operacji `userdirectory.update`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`

Przykłady

Pobieranie katalogów użytkowników

Pobierz wszystkie katalogi użytkowników z dodatkowymi właściwościami wyświetlającymi mapowania typów mediów oraz mapowania grup provisioningu powiązane z każdym katalogiem użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.get",
    "params": {
        "output": "extend",
        "selectProvisionMedia": "extend",
        "selectProvisionGroups": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Grupa użytkowników

Źródło

CUserDirectory::get() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.test

Opis

`object userdirectory.test(array userDirectory)`

Ta metoda umożliwia przetestowanie ustawień połączenia z katalogiem użytkowników.

Note:
Ta metoda umożliwia również sprawdzenie, jakie skonfigurowane dane odpowiadają ustawieniom katalogu użytkowników dla aprowizacji użytkowników (np. jaka rola użytkownika, grupy użytkowników, media użytkownika zostaną przypisane do użytkownika). W przypadku tego typu testu żądanie API powinno zostać wykonane dla katalogu użytkowników, dla którego `provision_status` ma ustawioną wartość enabled.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*.

Parametry

`(object)` Właściwości katalogu użytkowników.

Ponieważ API `userdirectory.get` nie zwraca pola `bind_password`, należy podać `userdirectoryid` i/lub `bind_password`.
Oprócz standardowych właściwości katalogu użytkowników, metoda akceptuje następujące parametry.

Parametr	Type	Opis
test_username	string	Nazwa użytkownika do przetestowania w katalogu użytkowników.
test_password	string	Hasło powiązane z nazwą użytkownika do przetestowania w katalogu użytkowników.

Zwracane wartości

`(bool)` Zwraca true w przypadku powodzenia.

Przykłady

Testuj katalog użytkowników dla istniejącego użytkownika

Przetestuj katalog użytkowników „3” dla „user1”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Testuj katalog użytkowników dla nieistniejącego użytkownika

Przetestuj katalog użytkowników „3” dla nieistniejącego użytkownika „user2”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "error": {
        "code": -32500,
        "message": "Application error.",
        "data": "Incorrect user name or password or account is temporarily blocked."
    },
    "id": 1
}
```

Testuj katalog użytkowników na potrzeby provisioningu użytkowników

Przetestuj katalog użytkowników „3”, aby sprawdzić, które skonfigurowane dane odpowiadają ustawieniom katalogu użytkowników dla provisioningu użytkownika „user3” (np. jaka rola użytkownika, grupy użytkowników i media użytkownika zostaną przypisane temu użytkownikowi).

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "username": "user3",
        "name": "John",
        "surname": "Doe",
        "medias": [],
        "usrgrps": [
```

```
            {
                "usrgrpid": "8"
            },
            {
                "usrgrpid": "7"
            }
        ],
        "roleid": "2",
        "userdirectoryid": "2"
    },
    "id": 1
}
```

Źródło

CUserDirectory::test() w *ui/include/classes/api/services/CUserDirectory.php*.

userdirectory.update

Opis

`object userdirectory.update(object/array userDirectory)`

Ta metoda umożliwia aktualizację istniejących katalogów użytkowników.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*.

Parametry

`(object/array)` Właściwości katalogu użytkowników, które mają zostać zaktualizowane.

Właściwość `userdirectoryid` musi być zdefiniowana dla każdego katalogu użytkowników, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych katalogów użytkowników we właściwości `userdirectoryids`.

Przykłady

Aktualizacja hasła bind dla katalogu użytkowników

Ustaw nowe hasło bind dla katalogu użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "userdirectory.update",
    "params": {
        "userdirectoryid": "3",
        "bind_password": "newldappassword"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "3"
        ]
    },
    "id": 1
}
```

Zaktualizuj mapowania dla katalogu użytkowników

Zaktualizuj mapowania grup udostępniania oraz mapowania typów mediów dla katalogu użytkowników „2”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userdirectoryids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CUserDirectory::update() w *ui/include/classes/api/services/CUserDirectory.php*.

Konfiguracja

Ta klasa jest przeznaczona do eksportowania i importowania danych konfiguracji Zabbix.

Dostępne metody:

- configuration.export - eksport konfiguracji
- configuration.import - import konfiguracji
- configuration.importcompare - porównanie pliku importu z bieżącymi elementami systemu

configuration.export

Opis

`string configuration.export(object parameters)`

Ta metoda umożliwia eksport danych konfiguracyjnych jako serializowanego ciągu znaków.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące obiekty do wyeksportowania oraz format, który ma zostać użyty.

Parameter	Type	Description
format	string	Format, w jakim dane muszą zostać wyeksportowane. Możliwe wartości: `yaml` - YAML; `xml` - XML; `json` - JSON; `raw` - nieprzetworzona tablica PHP. Zachowanie parametru: - *required*
prettyprint	boolean	Sprawia, że wynik jest bardziej czytelny dla człowieka poprzez dodanie wcięć. Możliwe wartości: `true` - dodaj wcięcia; `false` - *(domyślnie)* nie dodawaj wcięć.
options	object	Obiekty do wyeksportowania. Obiekt `options` ma następujące parametry: `host_groups` - `(array)` identyfikatory grup hostów do wyeksportowania; `hosts` - `(array)` identyfikatory hostów do wyeksportowania; `images` - `(array)` identyfikatory obrazów do wyeksportowania; `maps` - `(array)` identyfikatory map do wyeksportowania; `mediaTypes` - `(array)` identyfikatory typów mediów do wyeksportowania; `template_groups` - `(array)` identyfikatory grup szablonów do wyeksportowania; `templates` - `(array)` identyfikatory szablonów do wyeksportowania; `dashboards` - `(array)` identyfikatory pulpitów do wyeksportowania; `global_regexes` - `(array)` identyfikatory globalnych wyrażeń regularnych do wyeksportowania. Użytkownicy typu *Admin* i *User* mogą eksportować tylko te obiekty, dla których mają uprawnienia *read-only* lub *read-write* permission, a także obrazy, ale nie typy mediów. Zachowanie parametru: - *required*.

Zwracane wartości

`(string)` Zwraca serializowany ciąg znaków zawierający żądane dane konfiguracyjne.

Przykłady

Eksportowanie szablonu

Wyeksportuj konfigurację szablonu „10571” jako ciąg XML.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "configuration.export",
    "params": {
        "options": {
            "templates": [
                "10571"
            ]
        },
        "format": "xml"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>8.0</version><template_
    "id": 1
}
```

Źródło

CConfiguration::export() w *ui/include/classes/api/services/CConfiguration.php*.

configuration.import

Opis

`boolean configuration.import(object parameters)`

Ta metoda umożliwia import danych konfiguracyjnych z serializowanego ciągu znaków.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające dane do importu oraz reguły określające sposób obsługi tych danych.

Parameter	Type	Description
format	string	Format serializowanego ciągu. Możliwe wartości: `yaml` - YAML; `xml` - XML; `json` - JSON. Zachowanie parametru: - *required*
source	string	Serializowany ciąg zawierający dane konfiguracyjne. Zachowanie parametru: - *required*

Parameter	Type	Description
rules	object	Reguły określające sposób importu nowych i istniejących obiektów. Użytkownicy typu *Admin* mogą importować tylko te obiekty, dla których mają uprawnienia *read-write* permission, a także mapy. Na przykład host i jego encje (pozycje, wyzwalacze, wykresy itp.) mogą zostać zaimportowane tylko wtedy, gdy grupa użytkowników danego użytkownika ma uprawnienia do grupy hostów, do której będzie należeć importowany host. Obrazy i typy mediów nie mogą być importowane przez użytkowników typu *Admin*. Parametr `rules` jest szczegółowo opisany w tabeli poniżej. Zachowanie parametru: - *required*
returnMissingObjects	boolean	Reguły określające, czy obiekt odpowiedzi ma zawierać informacje o brakujących obiektach; domyślnie: `false`.

Attention:
Jeśli nie zostaną podane żadne reguły, konfiguracja nie zostanie zaktualizowana.

Obiekt `rules` obsługuje następujące parametry.

Parameter	Type	Description
dashboards	object	Reguły importu dashboardów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe dashboardy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true` i dany importowalny dashboard jest obecny w systemie, zostanie on w pełni zaktualizowany. Strony i widżety zostaną całkowicie zastąpione nowymi; domyślnie: `false`.
discoveryRules	object	Reguły importu reguł LLD. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe reguły LLD; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące reguły LLD zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, reguły LLD nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
global_regexes	object	Reguły importu globalnych wyrażeń regularnych. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe globalne wyrażenia regularne; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące globalne wyrażenia regularne zostaną zaktualizowane, a ich podwyrażenia zostaną całkowicie nadpisane; domyślnie: `false`.

Parameter	Type	Description
graphs	object	Reguły importu wykresów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe wykresy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wykresy zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wykresy nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
host_groups	object	Reguły importu grup hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe grupy hostów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy hostów zostaną zaktualizowane; domyślnie: `false`.
hosts	object	Reguły importu hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe hosty; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące hosty zostaną zaktualizowane; domyślnie: `false`.
httptests	object	Reguły importu scenariuszy web. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe scenariusze web; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące scenariusze web zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, scenariusze web nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
images	object	Reguły importu obrazów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe obrazy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące obrazy zostaną zaktualizowane; domyślnie: `false`.
items	object	Reguły importu pozycji. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe pozycje; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące pozycje zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, pozycje nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
maps	object	Reguły importu map. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe mapy; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy zostaną zaktualizowane; domyślnie: `false`.

Parameter	Type	Description
mediaTypes	object	Reguły importu typów mediów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe typy mediów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące typy mediów zostaną zaktualizowane; domyślnie: `false`.
templateDashboards	object	Reguły importu dashboardów szablonu. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe dashboardy szablonu; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące dashboardy szablonu zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, dashboardy szablonu nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
template_groups	object	Reguły importu grup szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe grupy szablonów; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy szablonów zostaną zaktualizowane; domyślnie: `false`.
templateLinkage	object	Reguły importu powiązań szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, szablony, które nie są powiązane z hostem lub szablonem będącym przedmiotem importu, ale są obecne w importowanych danych, zostaną powiązane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, szablony powiązane z hostem lub szablonem będącym przedmiotem importu, ale nieobecne w importowanych danych, zostaną odłączone bez usuwania encji (pozycji, wyzwalaczy itp.) odziedziczonych po odłączonych szablonach; domyślnie: `false`.
templates	object	Reguły importu szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe szablony; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące szablony zostaną zaktualizowane; domyślnie: `false`.
triggers	object	Reguły importu wyzwalaczy. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, zostaną utworzone nowe wyzwalacze; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wyzwalacze zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wyzwalacze nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Parameter	Type	Description
valueMaps	object	Reguły importu map wartości hosta lub szablonu. Obsługiwane parametry: `createMissing` - (`boolean`) jeśli ustawione na `true`, zostaną utworzone nowe mapy wartości; domyślnie: `false`; `updateExisting` - (`boolean`) jeśli ustawione na `true`, istniejące mapy wartości zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - (`boolean`) jeśli ustawione na `true`, mapy wartości nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Zwracane wartości

(`boolean`) Zwraca `true` jeżeli importowanie się powiodło.

Przykłady

Importowanie szablonu

Zaimportuj konfigurację szablonu zawartą w ciągu XML. Jeśli w ciągu XML brakuje jakichkolwiek pozycji lub wyzwalaczy, zostaną one usunięte z bazy danych, a wszystko pozostałe pozostanie bez zmian.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```

Źródło

CConfiguration::import() w *ui/include/classes/api/services/CConfiguration.php*.

configuration.importcompare

Opis

`array configuration.importcompare(object parameters)`

Ta metoda umożliwia porównanie pliku importu z bieżącymi elementami systemu i pokazuje, co zostanie zmienione po zaimportowaniu tego pliku importu.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające możliwe dane do zaimportowania oraz reguły określające, jak dane powinny być obsługiwane.

Parameter	Type	Description
format	string	Format serializowanego ciągu znaków. Możliwe wartości: `yaml` - YAML; `xml` - XML; `json` - JSON. Zachowanie parametru: - *required*
source	string	Serializowany ciąg znaków zawierający dane konfiguracyjne. Zachowanie parametru: - *required*
rules	object	Reguły określające, jak należy porównywać nowe i istniejące obiekty. Użytkownicy typu *Admin* i *User* mogą porównywać nowe obiekty tylko z istniejącymi, dla których mają uprawnienia *read-only* lub *read-write* permission. Parametr `rules` jest szczegółowo opisany w tabeli poniżej. Zachowanie parametru: - *required*
returnMissingObjects	boolean	Reguły określające, czy obiekt odpowiedzi powinien zawierać informacje o brakujących obiektach; domyślnie: `false`.

Attention:
Jeśli nie zostaną podane żadne reguły, nie będzie nic do zaktualizowania, a wynik będzie pusty.

Note:
Porównanie będzie wykonywane tylko dla grup hostów i szablonów. Wyzwalacze i wykresy będą porównywane tylko dla importowanych szablonów, a wszystkie pozostałe będą uznawane za ”new”.

Obiekt `rules` obsługuje następujące parametry.

Parameter	Type	Description
dashboards	object	Reguły określające sposób importu dashboardów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe dashboardy zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true` i dany importowalny dashboard jest obecny w systemie, zostanie on w pełni zaktualizowany. Strony i widżety zostaną całkowicie zastąpione nowymi; domyślnie: `false`.
discoveryRules	object	Reguły określające sposób importu reguł LLD. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe reguły LLD zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące reguły LLD zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, reguły LLD nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
global_regexes	object	Reguły określające sposób importu globalnych wyrażeń regularnych. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe globalne wyrażenia regularne zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące globalne wyrażenia regularne zostaną zaktualizowane, a ich podwyrażenia zostaną całkowicie nadpisane; domyślnie: `false`.
graphs	object	Reguły określające sposób importu wykresów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe wykresy zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wykresy zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wykresy nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
host_groups	object	Reguły określające sposób importu grup hostów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe grupy hostów zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy hostów zostaną zaktualizowane; domyślnie: `false`.
httptests	object	Reguły określające sposób importu scenariuszy web. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe scenariusze web zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące scenariusze web zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, scenariusze web nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Parameter	Type	Description
images	object	Reguły określające sposób importu obrazów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe obrazy zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące obrazy zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie ma wpływu na wynik. Jest dozwolony wyłącznie dla zachowania spójności z `configuration.import`.
items	object	Reguły określające sposób importu pozycji. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe pozycje zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące pozycje zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, pozycje nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
maps	object	Reguły określające sposób importu map. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe mapy zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie ma wpływu na wynik. Jest dozwolony wyłącznie dla zachowania spójności z `configuration.import`.
mediaTypes	object	Reguły określające sposób importu typów mediów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe typy mediów zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące typy mediów zostaną zaktualizowane; domyślnie: `false`. Ten parametr nie ma wpływu na wynik. Jest dozwolony wyłącznie dla zachowania spójności z `configuration.import`.
templateDashboards	object	Reguły określające sposób importu dashboardów szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe dashboardy szablonów zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące dashboardy szablonów zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, dashboardy szablonów nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
template_groups	object	Reguły określające sposób importu grup szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe grupy szablonów zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące grupy szablonów zostaną zaktualizowane; domyślnie: `false`.

Parameter	Type	Description
templateLinkage	object	Reguły określające sposób importu powiązań szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, szablony, które nie są powiązane z importowanym hostem lub szablonem, ale są obecne w importowanych danych, zostaną powiązane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, szablony powiązane z importowanym hostem lub szablonem, ale nieobecne w importowanych danych, zostaną odłączone bez usuwania encji (pozycji, wyzwalaczy itp.) odziedziczonych po odłączonych szablonach; domyślnie: `false`.
templates	object	Reguły określające sposób importu szablonów. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe szablony zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące szablony zostaną zaktualizowane; domyślnie: `false`.
triggers	object	Reguły określające sposób importu wyzwalaczy. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe wyzwalacze zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące wyzwalacze zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, wyzwalacze nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.
valueMaps	object	Reguły określające sposób importu map wartości hosta lub szablonu. Obsługiwane parametry: `createMissing` - `(boolean)` jeśli ustawione na `true`, nowe mapy wartości zostaną utworzone; domyślnie: `false`; `updateExisting` - `(boolean)` jeśli ustawione na `true`, istniejące mapy wartości zostaną zaktualizowane; domyślnie: `false`; `deleteMissing` - `(boolean)` jeśli ustawione na `true`, mapy wartości nieobecne w importowanych danych zostaną usunięte z bazy danych; domyślnie: `false`.

Zwracane wartości

`(array)` Zwraca tablicę ze zmianamy w konfiguracji, które zostaną wprowadzone.

Przykłady

Porównanie importu szablonu

Porównaj szablon zawarty w ciągu XML z bieżącymi elementami systemu i pokaż, co zostanie zmienione, jeśli ten szablon zostanie zaimportowany.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```


Źródło

CConfiguration::importcompare() w *ui/include/classes/api/services/CConfiguration.php*.

Konserwacja

Ta klasa jest przeznaczona do pracy z okresami konserwacji.

Odwołania do obiektów:

- Konserwacja
- Okres czasu
- Tag problemu

Dostępne metody:

- maintenance.create - utwórz nowe okresy konserwacji
- maintenance.delete - usuń okresy konserwacji
- maintenance.get - pobierz okresy konserwacji
- maintenance.update - zaktualizuj okresy konserwacji

Obiekt konserwacji

Następujące obiekty są bezpośrednio powiązane z API `maintenance`.

Konserwacja

Obiekt konserwacji ma następujące właściwości.

Property	Type	Description
maintenanceid	ID	ID konserwacji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa konserwacji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Property	Type	Description
active_since	timestamp	Czas, od którego konserwacja staje się aktywna (włącznie). Podana wartość zostanie zaokrąglona w dół do minut. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
active_till	timestamp	Czas, do którego konserwacja pozostaje aktywna (wyłącznie). Podana wartość zostanie zaokrąglona w dół do minut. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	string	Opis konserwacji.
maintenance_type	integer	Typ konserwacji. Możliwe wartości: 0 - *(domyślnie)* z pobieraniem danych; 1 - bez pobierania danych.
tags_evaltype	integer	Metoda oceny tagów problemu. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.

Okres czasu

Obiekt okresu czasu służy do definiowania okresów, w których konserwacja ma wejść w życie. Ma następujące właściwości.

Property	Type	Description
period	integer	Czas trwania okresu konserwacji w sekundach. Podana wartość zostanie zaokrąglona w dół do pełnych minut. Możliwy zakres wartości: 300 - 86399940. Domyślnie: 3600.
timeperiod_type	integer	Typ okresu czasu. Możliwe wartości: 0 - *(domyślnie)* tylko jeden raz; 2 - codziennie; 3 - co tydzień; 4 - co miesiąc.
start_date	timestamp	Data, kiedy okres konserwacji ma wejść w życie. Podana wartość zostanie zaokrąglona w dół do pełnych minut. Domyślnie: bieżąca data. Zachowanie właściwości: - *obsługiwane*, jeśli `timeperiod_type` jest ustawione na ”one time only”
start_time	integer	Pora dnia, o której konserwacja rozpoczyna się, wyrażona w sekundach. Podana wartość zostanie zaokrąglona w dół do pełnych minut. Domyślnie: 0. Zachowanie właściwości: - *obsługiwane*, jeśli `timeperiod_type` jest ustawione na ”daily”, ”weekly” lub ”monthly”

Property	Type	Description
every	integer	Dla okresów dziennych i tygodniowych `every` określa interwały dni lub tygodni, z jakimi konserwacja ma wejść w życie. Wartość domyślna, jeśli `timeperiod_type` jest ustawione na ”daily” lub ”weekly”: 1. Dla okresów miesięcznych, gdy ustawiono `day`, właściwość `every` określa dzień miesiąca, w którym konserwacja ma wejść w życie. Wartość domyślna, jeśli `timeperiod_type` jest ustawione na ”monthly” i ustawiono `day`: 1. Dla okresów miesięcznych, gdy ustawiono `dayofweek`, właściwość `every` określa tydzień miesiąca, w którym konserwacja ma wejść w życie. Możliwe wartości, jeśli `timeperiod_type` jest ustawione na ”monthly” i ustawiono `dayofweek`: 1 - *(domyślnie)* pierwszy tydzień; 2 - drugi tydzień; 3 - trzeci tydzień; 4 - czwarty tydzień; 5 - ostatni tydzień. Zachowanie właściwości: - *obsługiwane*, jeśli `timeperiod_type` jest ustawione na ”daily”, ”weekly” lub ”monthly”
dayofweek	integer	Dni tygodnia, w których konserwacja ma wejść w życie. Możliwe wartości bitmapy: 1 - poniedziałek; 2 - wtorek; 4 - środa; 8 - czwartek; 16 - piątek; 32 - sobota; 64 - niedziela. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości bitmapy (na przykład 21 dla poniedziałku, środy i piątku). Zachowanie właściwości: - *wymagane*, jeśli `timeperiod_type` jest ustawione na ”weekly” lub jeśli `timeperiod_type` jest ustawione na ”monthly” i `day` nie jest ustawione
day	integer	Dzień miesiąca, w którym konserwacja ma wejść w życie. Zachowanie właściwości: - *wymagane*, jeśli `timeperiod_type` jest ustawione na ”monthly” i `dayofweek` nie jest ustawione

Property	Type	Description
month	integer	Miesiące, w których konserwacja ma wejść w życie. Możliwe wartości bitmapy: 1 - styczeń; 2 - luty; 4 - marzec; 8 - kwiecień; 16 - maj; 32 - czerwiec; 64 - lipiec; 128 - sierpień; 256 - wrzesień; 512 - październik; 1024 - listopad; 2048 - grudzień. To pole jest maską bitową; akceptowalna jest dowolna suma możliwych wartości bitmapy (na przykład 585 dla stycznia, kwietnia, lipca i października). Zachowanie właściwości: - *wymagane*, jeśli `timeperiod_type` jest ustawione na ”monthly”

Nazwa zdarzenia

Obiekt nazwy zdarzenia służy do określania, które problemy muszą zostać wyciszone po rozpoczęciu okresu utrzymania. Nazwy zdarzeń można określić tylko wtedy, gdy parametr `maintenance_type` obiektu Maintenance jest ustawiony na „with data collection”. Obiekt ma następujące właściwości.

Property	Type	Description
operator	integer	Operator warunku. Możliwe wartości: 2 - *(domyślnie)* Zawiera; 3 - Nie zawiera.
value	string	Wartość nazwy zdarzenia. Zachowanie właściwości: - *wymagane*

Tag problemu

Obiekt tagu problemu służy do określania, które problemy muszą zostać wyciszone po rozpoczęciu okresu utrzymania. Tagi można określać tylko wtedy, gdy parametr `maintenance_type` obiektu Maintenance jest ustawiony na „with data collection”. Obiekt ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa tagu problemu. Zachowanie właściwości: - *required*
operator	integer	Warunek operatora. Możliwe wartości: 0 - Równa się; 1 - Nie równa się; 2 - *(default)* Zawiera; 3 - Nie zawiera.
value	string	Wartość tagu problemu.

maintenance.create

Opis

`object maintenance.create(object/array maintenances)`

Ta metoda umożliwia tworzenie nowych konserwacji.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Okresy konserwacji do utworzenia.

Oprócz standardowych właściwości okresu konserwacji metoda przyjmuje następujące parametry.

Parameter	Type	Description
groups	object/array	Grupy hostów, które mają zostać objęte okresem konserwacji. Grupy hostów mogą mieć zdefiniowaną wyłącznie właściwość `groupid`. Zachowanie parametru: - *wymagany*, jeśli nie ustawiono `hosts`
hosts	object/array	Hosty, które mają zostać objęte okresem konserwacji. Hosty mogą mieć zdefiniowaną wyłącznie właściwość `hostid`. Zachowanie parametru: - *wymagany*, jeśli nie ustawiono `groups`
triggers	object/array	Wyzwalacze, które mają zostać objęte okresem konserwacji. Wyzwalacze mogą mieć zdefiniowaną wyłącznie właściwość `triggerid`. Zachowanie parametru: - *wymagany*, jeśli nie ustawiono `groups` ani `hosts` - *obsługiwany*, jeśli maintenance_type=0 (z gromadzeniem danych)
event_names	object/array	Nazwy zdarzeń definiujące problemy, które mają zostać objęte okresem konserwacji. Zachowanie parametru: - *obsługiwany*, jeśli maintenance_type=0 (z gromadzeniem danych)
tags	object/array	Znaczniki problemów definiujące problemy, które mają zostać objęte okresem konserwacji. Zachowanie parametru: - *obsługiwany*, jeśli właściwość `maintenance_type` obiektu Maintenance jest ustawiona na „z gromadzeniem danych”
timeperiods	object/array	Okresy konserwacji. Zachowanie parametru: - *wymagany*

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych konserwacji w ramach właściwości `maintenanceids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych konserwacji.

Przykłady

Tworzenie okresu utrzymania

Utwórz okres utrzymania ze zbieraniem danych dla grupy hostów o identyfikatorze „20” oraz z tagami problemu **service:mysqld** i **error**.
Musi być aktywny od 17.03.2026 do 17.03.2027, rozpoczynać się w każdą niedzielę o 18:00 i trwać godzinę.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "3"
        ]
    },
    "id": 1
}
```

Tworzenie konserwacji ad hoc

Utwórz jednogodzinną konserwację ad hoc ze zbieraniem danych dla grupy hostów o identyfikatorze „4”, wyzwalacza o identyfikatorze „12345” oraz nazwy zdarzenia **Printer: ink low**.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "maintenanceids": [
            "4"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host
- Grupa hostów
- Wyzwalacz
- Nazwa zdarzenia
- Tag problemu
- Okres

Źródło

CMaintenance::create() w *ui/include/classes/api/services/CMaintenance.php*.

maintenance.delete

Opis

`object maintenance.delete(array maintenanceIds)`

Ta metoda umożliwia usuwanie okresów konserwacji.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` Identyfikatory okresów konserwacji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych okresów konserwacji we właściwości `maintenanceids`.

Przykłady

Usuwanie wielu okresów konserwacji

Usuń dwa okresy konserwacji.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "maintenance.delete",
"params": [
"3",
"1"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"maintenanceids": [
"3",
"1"
]
},
"id": 1
}
```

Źródło

CMaintenance::delete() w *ui/include/classes/api/services/CMaintenance.php*.

maintenance.get

Opis

`integer/array maintenance.get(object parameters)`

Metoda umożliwia pobieranie konserwacji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwraca tylko okresy utrzymania przypisane do podanych grup hostów.
hostids	ID/array	Zwraca tylko okresy utrzymania przypisane do podanych hostów.
triggerids	ID/array	Zwraca tylko okresy utrzymania przypisane do podanych wyzwalaczy.
maintenanceids	ID/array	Zwraca tylko okresy utrzymania o podanych identyfikatorach.
selectHostGroups	query	Zwraca właściwość `hostgroups` zawierającą grupy hostów przypisane do okresu utrzymania.
selectHosts	query	Zwraca właściwość `hosts` zawierającą hosty przypisane do okresu utrzymania.
selectTriggers	query	Zwraca właściwość `triggers` zawierającą wyzwalacze przypisane do okresu utrzymania.

Parametr	Typ	Opis
selectEventNames	query	Zwraca właściwość `event_names` zawierającą nazwy zdarzeń przypisane do okresu utrzymania.
selectTags	query	Zwraca właściwość `tags` zawierającą znaczniki problemów okresu utrzymania.
selectTimeperiods	query	Zwraca właściwość `timeperiods` zawierającą okresy czasowe okresu utrzymania.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `maintenanceid`, `name`, `maintenance_type`, `active_since`, `active_till`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie okresów utrzymania

Pobierz wszystkie skonfigurowane okresy utrzymania, dane dotyczące przypisanego zakresu okresu utrzymania (grupa hostów, host, wyzwalacz, nazwa zdarzenia, znaczniki) oraz zdefiniowane przedziały czasowe.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "maintenance.get",
    "params": {
        "output": "extend",
        "selectHostGroups": "extend",
        "selectHosts": "extend",
        "selectTriggers": "extend",
        "selectEventNames": "extend",
        "selectTags": "extend",
        "selectTimeperiods": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "maintenanceid": "1",
      "name": "Regular DB maintenance",
      "maintenance_type": "0",
      "description": "",
      "active_since": "1791936000",
```

```
      "active_till": "2107555200",
      "tags_evaltype": "0",
      "hostgroups": [
        {
          "groupid": "20",
          "name": "Databases",
          "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: Printer: ink OUT",
      "maintenance_type": "0",
      "description": "",
      "active_since": "1792057740",
      "active_till": "1792061340",
      "tags_evaltype": "0",
      "hostgroups": [
        {
          "groupid": "4",
          "name": "Zabbix servers",
          "flags": "0",
          "uuid": "6f6799aa69e844b4b3918f779f2abf08"
        }
      ],
      "triggers": [],
      "event_names": [
        {
          "operator": "2",
          "value": "Printer: ink OUT"
        }
      ],
      "hosts": [],
      "tags": [],
      "timeperiods": [
        {
          "timeperiod_type": "0",
          "every": "1",
          "month": "0",
          "dayofweek": "0",
          "day": "0",
          "start_time": "0",
          "period": "3600",
          "start_date": "1792057740"
```

```
        }
      ]
    }
  ],
  "id": 1
}
```

Zobacz także

- Host
- Grupa hostów
- Wyzwalacz
- Nazwa zdarzenia
- Tag problemu
- Okres

Źródło

CMaintenance::get() w *ui/include/classes/api/services/CMaintenance.php*.

maintenance.update

Opis

`object maintenance.update(object/array maintenances)`

Ta metoda umożliwia aktualizację istniejących konserwacji.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości konserwacji do zaktualizowania.

Właściwość `maintenanceid` musi być zdefiniowana dla każdej konserwacji, a wszystkie pozostałe właściwości są opcjonalne. Zostaną zaktualizowane tylko przekazane właściwości, a wszystkie pozostałe pozostaną niezmienione.

Oprócz standardowych właściwości konserwacji metoda akceptuje następujące parametry.

Parameter	Type	Description
groups	object/array	Grupy hostów zastępujące bieżące grupy. Grupy hostów mogą mieć zdefiniowaną tylko właściwość `groupid`. Zachowanie parametru: - *wymagany*, jeśli nie ustawiono `hosts`
hosts	object/array	Hosty zastępujące bieżące hosty. Hosty mogą mieć zdefiniowaną tylko właściwość `hostid`. Zachowanie parametru: - *wymagany*, jeśli nie ustawiono `groups`
triggers	object/array	Wyzwalacze zastępujące bieżące wyzwalacze. Wyzwalacze mogą mieć zdefiniowaną tylko właściwość `triggerid`. Zachowanie parametru: - *wymagany*, jeśli nie ustawiono `groups` ani `hosts` - *obsługiwany*, jeśli maintenance_type=0 (ze zbieraniem danych)
event_names	object/array	Nazwy zdarzeń zastępujące bieżące nazwy zdarzeń. Zachowanie parametru: - *obsługiwany*, jeśli maintenance_type=0 (ze zbieraniem danych)

Parameter	Type	Description
tags	object/array	Znaczniki problemów zastępujące bieżące znaczniki. Zachowanie parametru: - *obsługiwany*, jeśli właściwość `maintenance_type` obiektu konserwacji jest ustawiona na „ze zbieraniem danych”
timeperiods	object/array	Okresy konserwacji zastępujące bieżące okresy.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych konserwacji w ramach właściwości `maintenanceids`.

Przykłady

Przypisywanie różnych hostów

Zastąp hosty aktualnie przypisane do konserwacji dwoma innymi.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "maintenance.update",
"params": {
"maintenanceid": "3",
"hosts": [
{"hostid": "10085"},
{"hostid": "10084"}
]
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"maintenanceids": [
"3"
]
},
"id": 1
}
```

Zobacz także

- Okres czasu

Źródło

CMaintenance::update() w *ui/include/classes/api/services/CMaintenance.php*.

Kontrola wykrywania

Ta klasa została zaprojektowana do pracy z kontrolami wykrywania.

Odwołania do obiektów:

- Kontrola wykrywania

Dostępne metody:

- dcheck.get - pobieranie kontroli wykrywania

Obiekt sprawdzania odkrycia

Następujące obiekty są bezpośrednio powiązane z API `dcheck`.

Kontrola wykrywania

Obiekt kontroli wykrywania definiuje konkretną kontrolę wykonywaną przez regułę wykrywania sieci. Ma następujące właściwości.

Właściwość	Typ	Opis
dcheckid	ID	ID kontroli wykrywania.
druleid	ID	ID reguły wykrywania, do której należy ta kontrola.
key_	string	Klucz pozycji (jeśli `type` jest ustawione na ”Zabbix agent”) lub OID SNMP (jeśli `type` jest ustawione na ”SNMPv1 agent”, ”SNMPv2 agent” lub ”SNMPv3 agent”). Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent” lub ”SNMPv3 agent”
ports	string	Jeden lub kilka zakresów portów do sprawdzenia, oddzielonych przecinkami. Domyślnie: 0. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”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) lub ”Telnet” (15)
snmp_community	string	Społeczność SNMP. Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”SNMPv1 agent” lub ”SNMPv2 agent”
snmpv3_authpassphrase	string	Hasło uwierzytelniania. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
snmpv3_authprotocol	integer	Protokół uwierzytelniania. Możliwe wartości: 0 - *(domyślnie)* MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
snmpv3_contextname	string	Nazwa kontekstu SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent”
snmpv3_privpassphrase	string	Hasło prywatności. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authPriv”

Właściwość	Typ	Opis
snmpv3_privprotocol	integer	Protokół prywatności. Możliwe wartości: 0 - *(domyślnie)* DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent” i `snmpv3_securitylevel` jest ustawione na ”authPriv”
snmpv3_securitylevel	string	Poziom bezpieczeństwa. Możliwe wartości: 0 - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent”
snmpv3_securityname	string	Nazwa bezpieczeństwa. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”SNMPv3 agent”
type	integer	Typ kontroli. Możliwe wartości: 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. Zachowanie właściwości: - *wymagane*
uniq	integer	Czy używać tej kontroli jako kryterium unikalności urządzenia. Dla reguły wykrywania można skonfigurować tylko jedną unikalną kontrolę. Możliwe wartości: 0 - *(domyślnie)* nie używaj tej kontroli jako kryterium unikalności; 1 - użyj tej kontroli jako kryterium unikalności. Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”Zabbix agent”, ”SNMPv1 agent”, ”SNMPv2 agent” lub ”SNMPv3 agent”

Właściwość	Typ	Opis
host_source	integer	Źródło nazwy hosta. Możliwe wartości: 1 - *(domyślnie)* DNS; 2 - IP; 3 - wartość wykrywania tej kontroli.
name_source	integer	Źródło widocznej nazwy. Możliwe wartości: 0 - *(domyślnie)* nie określono; 1 - DNS; 2 - IP; 3 - wartość wykrywania tej kontroli.
allow_redirect	integer	Zezwalaj na sytuację, w której cel sprawdzany za pomocą ICMP ping odpowiada z innego adresu IP. Możliwe wartości: 0 - *(domyślnie)* traktuj przekierowane odpowiedzi tak, jakby docelowy host był niedostępny (niepowodzenie); 1 - traktuj przekierowane odpowiedzi tak, jakby docelowy host był dostępny (sukces). Zachowanie właściwości: - *obsługiwane* jeśli `type` jest ustawione na ”ICMP ping”

dcheck.get

Opis

`integer/array dcheck.get(object parameters)`

Metoda umożliwia pobieranie kontroli wykrywania zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dcheckids	ID/array	Zwraca tylko kontrole wykrywania o podanych identyfikatorach.
druleids	ID/array	Zwraca tylko kontrole wykrywania należące do podanych reguł wykrywania.
dserviceids	ID/array	Zwraca tylko kontrole wykrywania, które wykryły podane wykryte usługi.
selectDRules	query	Zwraca reguły wykrywania powiązane z kontrolami wykrywania.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dcheckid`, `druleid`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	

Parametr	Typ	Opis
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę otrzymanych obiektów, jeżeli parametr `countOutput` został użyty.

Przykłady

Pobieranie testów wykrywania dla reguły wykrywania

Pobierz wszystkie testy wykrywania, które są używane przez regułę ”6”.

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "dcheck.get",
  "params": {
    "output": "extend",
    "dcheckids": "6"
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "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
}
```

Źródło

CDCheck::get() w *ui/include/classes/api/services/CDCheck.php*.

Korelacja

Ta klasa jest przeznaczona do pracy z korelacjami.

Odwołania do obiektów:

- Korelacja
- Operacja korelacji
- Filtr korelacji
  - Warunek filtra korelacji

Dostępne metody:

- correlation.create - tworzenie nowych korelacji
- correlation.delete - usuwanie korelacji
- correlation.get - pobieranie korelacji
- correlation.update - aktualizowanie korelacji

Obiekt Correlation =

Następujące obiekty są bezpośrednio powiązane z API `correlation`.

Korelacja

Obiekt korelacji ma następujące właściwości.

Właściwość	Typ	Opis
correlationid	ID	ID korelacji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa korelacji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	string	Opis korelacji.
status	integer	Określa, czy korelacja jest włączona czy wyłączona. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.

Operacja korelacji

Obiekt operacji korelacji definiuje operację, która zostanie wykonana po uruchomieniu korelacji. Ma następujące właściwości.

Właściwość	Type	Opis
type	integer	Typ operacji. Możliwe wartości: 0 - zamknij stare zdarzenia; 1 - zamknij nowe zdarzenie. Zachowanie właściwości: - *wymagane*

Filtr korelacji

Obiekt filtra korelacji definiuje zestaw warunków, które muszą zostać spełnione, aby wykonać skonfigurowane operacje korelacji. Ma następujące właściwości.

Property	Type	Description
conditions	array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich umieszczenia we wzorze. Zachowanie właściwości: - *required*
evaltype	integer	Metoda oceny warunku filtra. Possible values: 0 - And/Or; 1 - And; 2 - Or; 3 - Custom expression. Zachowanie właściwości: - *required*
eval_formula	string	Wygenerowane wyrażenie, które zostanie użyte do oceny warunków filtra. Wyrażenie zawiera identyfikatory odwołujące się do konkretnych warunków filtra za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *read-only*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do konkretnych warunków filtra za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *required* jeśli `evaltype` jest ustawione na ”custom expression”

Warunek filtra korelacji

Obiekt warunku filtra korelacji definiuje określony warunek, który musi zostać sprawdzony przed uruchomieniem operacji korelacji.

Property	Type	Description
type	integer	Typ warunku. Możliwe wartości: 0 - tag starego zdarzenia; 1 - tag nowego zdarzenia; 2 - grupa hostów nowego zdarzenia; 3 - para tagów zdarzenia; 4 - wartość tagu starego zdarzenia; 5 - wartość tagu nowego zdarzenia. Zachowanie właściwości: - *wymagane*
tag	string	Tag zdarzenia (starego lub nowego). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „tag starego zdarzenia”, „tag nowego zdarzenia”, „wartość tagu starego zdarzenia” lub „wartość tagu nowego zdarzenia”

Property	Type	Description
groupid	ID	ID grupy hostów. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „grupa hostów nowego zdarzenia”
oldtag	string	Tag starego zdarzenia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „para tagów zdarzenia”
newtag	string	Tag starego zdarzenia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „para tagów zdarzenia”
value	string	Wartość tagu zdarzenia (starego lub nowego). Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „wartość tagu starego zdarzenia” lub „wartość tagu nowego zdarzenia”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu ich pobrania.
operator	integer	Operator warunku. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „grupa hostów nowego zdarzenia”, „wartość tagu starego zdarzenia” lub „wartość tagu nowego zdarzenia”

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod correlation.get i correlation.create.

Dla każdego typu warunku obsługiwane są następujące operatory i wartości.

Condition	Condition name	Supported operators	Expected value
2	Grupa hostów	=, <>	ID grupy hostów.
4	Wartość tagu starego zdarzenia	=, <>, like, not like	string
5	Wartość tagu nowego zdarzenia	=, <>, like, not like	string

correlation.create

Opis

`object correlation.create(object/array correlations)`

Ta metoda umożliwia tworzenie nowych korelacji.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Korelacje do utworzenia.

Oprócz standardowych właściwości korelacji, metoda akceptuje następujące parametry.

Parametr	Type	Opis
operations	array	Operacje korelacji do utworzenia dla korelacji. Zachowanie parametru: - *wymagany*
filter	object	Obiekt filtra korelacji dla korelacji. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID utworzonych korelacji we właściwości `correlationids`. Kolejność zwracanych ID są zgodne z kolejności przekazanych korelacji.

Przykłady

Utwórz nową korelację tagów zdarzeń

Utwórz korelację, używając metody oceny `AND/OR`, z jednym warunkiem i jedną operacją. Domyślnie korelacja będzie włączona.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Używanie niestandardowego filtra wyrażenia

Utwórz korelację, która będzie używać niestandardowego warunku filtra. Identyfikatory formuły „A” lub „B” zostały wybrane arbitralnie. Typ warunku będzie „Grupa hostów” z operatorem „<>”.

Żądanie:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "correlation.create",
    "params": {
        "name": "new host group correlation",
        "description": "a custom description",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr korelacji
- Operacja korelacji

Źródło

CCorrelation::create() w *ui/include/classes/api/services/CCorrelation.php*.

correlation.delete

Opis

`object correlation.delete(array correlationids)`

Ta metoda umożliwia usuwanie korelacji.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` IDs korelacji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający ID usuniętych korelacji we właściwości `correlationids`.

Przykład

Usuń wiele korelacji

Usuń dwie korelacje.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "correlation.delete",
"params": [
"1",
"2"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"correlationids": [
"1",
"2"
]
},
"id": 1
}
```

Źródło

CCorrelation::delete() w *ui/include/classes/api/services/CCorrelation.php*.

correlation.get

Opis

`integer/array correlation.get(object parameters)`

Metoda umożliwia pobieranie korelacji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
correlationids	ID/array	Zwraca tylko korelacje o podanych identyfikatorach.
selectFilter	query	Zwraca właściwość `filter` z warunkami korelacji.
selectOperations	query	Zwraca właściwość `operations` z operacjami korelacji.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `correlationid`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	

Parametr	Typ	Opis
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został użyty.

Przykłady

Pobieranie korelacji

Pobierz wszystkie skonfigurowane korelacje wraz z warunkami i operacjami korelacji. Filtr używa typu oceny „and/or”, dlatego właściwość `formula` jest pusta, a `eval_formula` jest generowana automatycznie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectFilter": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Filtr korelacji
- Operacja korelacji

Źródło

CCorrelation::get() w *ui/include/classes/api/services/CCorrelation.php*.

correlation.update

Opis

`object correlation.update(object/array correlations)`

Ta metoda umożliwia aktualizację istniejących korelacji.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości korelacji do zaktualizowania.

Właściwość `correlationid` musi być zdefiniowana dla każdej korelacji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości korelacji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Obiekt filtra korelacji zastępujący bieżący filtr.
operations	array	Operacje korelacji zastępujące istniejące operacje.

Zwracane wartości

`(object)` Zwraca obiekt zawierajacy identyfikatory zaktualizowanych korelacji we właściwości ‘correlationids.

Przykłady

Wyłącz korelację

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "status": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
```

```
    "id": 1
}
```

Zastąp warunki, ale zachowaj metodę oceny

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "correlation.update",
    "params": {
        "correlationid": "1",
        "filter": {
            "conditions": [
                {
                    "type": 3,
                    "oldtag": "error",
                    "newtag": "ok"
                }
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "correlationids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr korelacji
- Operacja korelacji

Źródło

CCorrelation::update() w *ui/include/classes/api/services/CCorrelation.php*.

Makro użytkownika

Ta klasa jest przeznaczona do pracy z makrami użytkownika na poziomie hosta oraz globalnymi makrami użytkownika.

Odwołania do obiektów:

- Makro globalne
- Makro hosta
    - Konfiguracja makra

Dostępne metody:

- usermacro.create - utwórz nowe makra hosta
- usermacro.createglobal - utwórz nowe makra globalne
- usermacro.delete - usuń makra hosta
- usermacro.deleteglobal - usuń makra globalne
- usermacro.get - pobierz makra hosta i makra globalne
- usermacro.update - zaktualizuj makra hosta
- usermacro.updateglobal - zaktualizuj makra globalne

Obiekt makra użytkownika

Następujące obiekty są bezpośrednio powiązane z API `usermacro`.

Makro globalne

Obiekt makra globalnego ma następujące właściwości.

Właściwość	Typ	Opis
globalmacroid	ID	ID makra globalnego. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	string	Opis makra.
macro	string	Ciąg makra. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Typ makra. Możliwe wartości: 0 - *(domyślnie)* Makro tekstowe; 1 - Makro sekretne; 2 - Sekret Vault.
value	string	Wartość makra. Zachowanie właściwości: - *tylko do zapisu*, jeśli `type` jest ustawione na „Makro sekretne” - *wymagane* dla operacji tworzenia

Makro hosta

Obiekt makra hosta definiuje makro dostępne na hoście, prototypie hosta lub szablonie. Ma następujące właściwości.

Właściwość	Typ	Opis
hostmacroid	ID	ID makra. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
automatic	integer	Określa, czy makro jest kontrolowane przez regułę wykrywania. Możliwe wartości: 0 - *(domyślnie)* Makro jest zarządzane przez użytkownika; 1 - Makro jest zarządzane przez regułę wykrywania. Użytkownik nie może tworzyć automatycznego makra. Aby zaktualizować automatyczne makro, musi ono zostać przekształcone na ręczne.
config	object/array	Konfiguracja makra odpowiedzialna za sposób wyświetlania makra w Kreatorze hosta.
description	string	Opis makra.
hostid	ID	ID hosta, prototypu hosta lub szablonu, do którego należy makro. Zachowanie właściwości: - *stała* - *wymagane* dla operacji tworzenia
macro	string	Ciąg makra. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Typ	Opis
type	integer	Typ makra. Możliwe wartości: 0 - *(domyślnie)* Makro tekstowe; 1 - Makro tajne; 2 - Sekret z magazynu.
value	string	Wartość makra. Zachowanie właściwości: - *tylko do zapisu*, jeśli `type` jest ustawione na „Makro tajne” - *wymagane* dla operacji tworzenia

Konfiguracja makr

Obiekt konfiguracji makr definiuje sposób wyświetlania makra w Kreatorze hosta.

Właściwość	Typ	Opis
type	integer	Typ pola wejściowego makra. Możliwe wartości: 0 - Makro nie jest używane w Kreatorze hosta; 1 - Pole tekstowe; 2 - Lista; 3 - Pole wyboru. Zachowanie właściwości: - *wymagane*
label	string	Etykieta pola wejściowego makra. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Pole tekstowe”, „Lista” lub „Pole wyboru”
description	text	Tekst pomocy wyświetlany obok pola wejściowego makra. Obsługuje formatowanie Markdown.
priority	integer	Pozycja pola wejściowego makra na liście makr. Jeśli `priority` nie jest ustawione, makro jest dodawane na końcu listy niepogrupowanych makr.
required	integer	Oznacza makro jako obowiązkowe. Możliwe wartości: 0 - Nieobowiązkowe; 1 - Obowiązkowe. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Pole tekstowe” lub „Lista”
regex	string	Wyrażenie regularne służące do walidacji danych wejściowych użytkownika w polu tekstowym. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Pole tekstowe”
section_name	string	Etykieta zwijanej sekcji, w której makro jest grupowane. Jeśli `section_name` nie jest ustawione, makro nie jest grupowane. Niepogrupowane makra są wyświetlane jako pierwsze; pogrupowane makra są wyświetlane poniżej, uporządkowane według ich `priority` w obrębie każdej zwijanej sekcji.

Właściwość	Typ	Opis
options	text	Ciąg JSON definiujący elementy listy lub wartości pola wyboru. Dla list: tablica obiektów z właściwościami `value` i `text`. Przykład: `[{"value": "http", "text": "HTTP"}, {"value": "https", "text": "HTTPS"}]` Dla pól wyboru: obiekt z właściwościami `checked` i `unchecked`. Przykład: `{"checked": true, "unchecked": false}` Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Lista” lub „Pole wyboru”

usermacro.create

Opis

`object usermacro.create(object/array hostMacros)`

Ta metoda umożliwia tworzenie nowych makr hosta.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Makra hosta do utworzenia.

Metoda akceptuje makra hosta ze standardowymi właściwościami makr hosta.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych makr hosta we właściwości `hostmacroids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych makr hosta.

Przykłady

Tworzenie makra hosta

Utwórz makro hosta `{$SNMP_COMMUNITY}` o wartości ”public” na hoście ”10198”.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.create",
    "params": {
        "hostid": "10198",
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "11"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::create() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.createglobal

Opis

`object usermacro.createglobal(object/array globalMacros)`

Ta metoda umożliwia tworzenie nowych globalnych makr.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Makra globalne do utworzenia.

Metoda akceptuje makra globalne ze standardowymi właściwościami makr globalnych.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych makr globalnych we właściwości `globalmacroids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych makr globalnych.

Przykłady

Tworzenie globalnego makra

Utwórz globalne makro `{$SNMP_COMMUNITY}` o wartości "public".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.createglobal",
    "params":  {
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::createGlobal() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.delete

Opis

`object usermacro.delete(array hostMacroIds)`

Ta metoda umożliwia usuwanie makr hosta.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID makr hosta do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych makr hosta we właściwości `hostmacroids`.

Przykłady

Usuwanie wielu makr hosta

Usuń dwa makra hosta.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "usermacro.delete",
"params": [
"32",
"11"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"hostmacroids": [
"32",
"11"
]
},
"id": 1
}
```

Źródło

CUserMacro::delete() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.deleteglobal

Opis

`object usermacro.deleteglobal(array globalMacroIds)`

Ta metoda umożliwia usuwanie globalnych makr.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID globalnych makr do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych globalnych makr we właściwości `globalmacroids`.

Przykłady

Usuwanie wielu makr globalnych

Usuń dwie makra globalne.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "usermacro.deleteglobal",
"params": [
"32",
"11"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"globalmacroids": [
"32",
"11"
]
},
"id": 1
}
```

Źródło

CUserMacro::deleteGlobal() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.get

Opis

`integer/array usermacro.get(object parameters)`

Metoda umożliwia pobieranie makr hosta i globalnych zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
globalmacro	flag	Zwraca makra globalne zamiast makr hosta.
globalmacroids	ID/array	Zwraca tylko makra globalne o podanych identyfikatorach.
groupids	ID/array	Zwraca tylko makra hosta należące do hostów lub szablonów z podanych grup hostów lub grup szablonów.
hostids	ID/array	Zwraca tylko makra należące do podanych hostów lub szablonów.
hostmacroids	ID/array	Zwraca tylko makra hosta o podanych identyfikatorach.
inherited	boolean	Jeśli ustawione na `true`, zwraca tylko makra użytkownika prototypu hosta dziedziczone z szablonu.
selectHostGroups	query	Zwraca grupy hostów, do których należy makro hosta, we właściwości `hostgroups`. Używane tylko podczas pobierania makr hosta.
selectHosts	query	Zwraca hosty, do których należy makro hosta, we właściwości `hosts`. Używane tylko podczas pobierania makr hosta.

Parametr	Typ	Opis
selectTemplateGroups	query	Zwraca grupy szablonów, do których należy makro szablonu, we właściwości `templategroups`. Używane tylko podczas pobierania makr szablonu.
selectTemplates	query	Zwraca szablony, do których należy makro hosta, we właściwości `templates`. Używane tylko podczas pobierania makr hosta.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `macro`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie makr hosta dla hosta

Pobierz wszystkie makra hosta zdefiniowane dla hosta „10198”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10198"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobieranie makr hosta dla szablonu

Pobierz makra hosta zdefiniowane dla szablonu „10265”, które zawierają „STATUS” w nazwie makra.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "hostids": "10265",
        "search": {
            "macro": "STATUS"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "hostmacroid": "6709",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.HOST}",
            "value": "",
            "description": "Nazwa hosta lub adres IP hosta strony statusu Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "1",
                "section_name": "",
                "label": "Host statusu Apache",
                "description": "Nazwa hosta lub adres IP hosta strony statusu Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "814",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.PATH}",
            "value": "server-status?auto",
            "description": "Ścieżka URL strony statusu Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "3",
```

```
                "section_name": "",
                "label": "Ścieżka strony statusu Apache",
                "description": "Ścieżka URL strony statusu Apache.",
                "required": "1",
                "regex": "",
                "options": []
            }
        },
        {
            "hostmacroid": "815",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.PORT}",
            "value": "80",
            "description": "Port strony statusu Apache.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "1",
                "priority": "2",
                "section_name": "",
                "label": "Port strony statusu Apache",
                "description": "W zakresie od 1 do 65535 włącznie.",
                "required": "1",
                "regex": "^-?([0-9]+|(([0-9]+)\\.([0-9]+)))$",
                "options": []
            }
        },
        {
            "hostmacroid": "816",
            "hostid": "10265",
            "macro": "{$APACHE.STATUS.SCHEME}",
            "value": "http",
            "description": "Schemat żądania, którym może być HTTP lub HTTPS.",
            "type": "0",
            "automatic": "0",
            "config": {
                "type": "2",
                "priority": "4",
                "section_name": "",
                "label": "Schemat żądania",
                "description": "Schemat żądania, którym może być HTTP lub HTTPS.",
                "required": "0",
                "regex": "",
                "options": [
                    {
                        "value": "http",
                        "text": "HTTP"
                    },
                    {
                        "value": "https",
                        "text": "HTTPS"
                    }
                ]
            }
        }
    ],
    "id": 1
}
```


Pobieranie makr globalnych

Pobierz wszystkie makra globalne.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.get",
    "params": {
        "output": "extend",
        "globalmacro": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "globalmacroid": "6",
            "macro": "{$SNMP_COMMUNITY}",
            "value": "public",
            "description": "",
            "type": "0"
        }
    ],
    "id": 1
}
```

Źródło

CUserMacro::get() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.update

Opis

`object usermacro.update(object/array hostMacros)`

Ta metoda umożliwia aktualizację istniejących makr hosta.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości makra hosta, które mają zostać zaktualizowane.

Właściwość `hostmacroid` musi być zdefiniowana dla każdego makra hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych makr hosta we właściwości `hostmacroids`.

Przykłady

Zmiana wartości makra hosta

Zmień wartość makra hosta na „public”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "public"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Zmień wartość makra utworzonego przez regułę wykrywania

Zmień utworzone przez regułę wykrywania makro „automatic” na „manual” i zmień jego wartość na „new-value”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "new-value",
        "automatic": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::update() w *ui/include/classes/api/services/CUserMacro.php*.

usermacro.updateglobal

Opis

`object usermacro.updateglobal(object/array globalMacros)`

Ta metoda umożliwia aktualizację istniejących globalnych makr.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości makra globalnego, które mają zostać zaktualizowane.

Właściwość `globalmacroid` musi być zdefiniowana dla każdego makra globalnego, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanego globalnego makra pod właściwością globalmacroids.

Przykłady

Zmiana wartości makra globalnego

Zmień wartość makra globalnego na „public”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "usermacro.updateglobal",
    "params": {
        "globalmacroid": "1",
        "value": "public"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CUserMacro::updateGlobal() in *ui/include/classes/api/services/CUserMacro.php*.

Mapa

Ta klasa jest przeznaczona do pracy z mapami.

Odwołania do obiektów:

- Mapa
- Element mapy
  - Element mapy Host
  - Element mapy Grupa hostów
  - Element mapy Mapa
  - Element mapy Wyzwalacz
  - Element mapy Tag
  - Element mapy URL
- Łącze mapy
  - Wyzwalacz łącza mapy
  - Wskaźniki łącza mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształty mapy
- Linie mapy

Dostępne metody:

- map.create - utwórz nowe mapy
- map.delete - usuń mapy
- map.get - pobierz mapy
- map.update - zaktualizuj mapy

Obiekt mapy

Następujące obiekty są bezpośrednio powiązane z API `map`.

Mapa

Obiekt mapy ma następujące właściwości.

Właściwość	Typ	Opis
sysmapid	ID	ID mapy. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
height	integer	Wysokość mapy w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa mapy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
width	integer	Szerokość mapy w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
backgroundid	ID	ID obrazu używanego jako tło mapy.
background_scale	integer	Czy włączyć proporcjonalne skalowanie obrazu tła. Możliwe wartości: 0 - wyłącz skalowanie; 1 - *(domyślnie)* włącz skalowanie obrazu.
expand_macros	integer	Czy rozwijać makra w etykietach podczas konfigurowania mapy. Możliwe wartości: 0 - *(domyślnie)* nie rozwijaj makr; 1 - rozwijaj makra.
expandproblem	integer	Czy wyzwalacz problemu będzie wyświetlany dla elementów z pojedynczym problemem. Możliwe wartości: 0 - zawsze wyświetlaj liczbę problemów; 1 - *(domyślnie)* wyświetlaj wyzwalacz problemu, jeśli istnieje tylko jeden problem.
grid_align	integer	Czy włączyć wyrównywanie do siatki. Możliwe wartości: 0 - wyłącz wyrównywanie do siatki; 1 - *(domyślnie)* włącz wyrównywanie do siatki.
grid_show	integer	Czy wyświetlać siatkę na mapie. Możliwe wartości: 0 - nie wyświetlaj siatki; 1 - *(domyślnie)* wyświetlaj siatkę.
grid_size	integer	Rozmiar siatki mapy w pikselach. Obsługiwane wartości: 20, 40, 50, 75 i 100. Domyślnie: 50.
highlight	integer	Czy podświetlanie ikon jest włączone. Możliwe wartości: 0 - podświetlanie wyłączone; 1 - *(domyślnie)* podświetlanie włączone.
iconmapid	ID	ID mapy ikon używanej na mapie.

Właściwość	Typ	Opis
label_format	integer	Czy włączyć zaawansowane etykiety. Możliwe wartości: 0 - *(domyślnie)* wyłącz zaawansowane etykiety; 1 - włącz zaawansowane etykiety.
label_location	integer	Położenie etykiety elementu mapy. Możliwe wartości: 0 - *(domyślnie)* dół; 1 - lewo; 2 - prawo; 3 - góra.
label_string_host	string	Niestandardowa etykieta dla elementów typu host. Zachowanie właściwości: - *wymagane*, jeśli `label_type_host` jest ustawione na ”custom”
label_string_hostgroup	string	Niestandardowa etykieta dla elementów grupy hostów. Zachowanie właściwości: - *wymagane*, jeśli `label_type_hostgroup` jest ustawione na ”custom”
label_string_image	string	Niestandardowa etykieta dla elementów obrazu. Zachowanie właściwości: - *wymagane*, jeśli `label_type_image` jest ustawione na ”custom”
label_string_map	string	Niestandardowa etykieta dla elementów mapy. Zachowanie właściwości: - *wymagane*, jeśli `label_type_map` jest ustawione na ”custom”
label_string_trigger	string	Niestandardowa etykieta dla elementów wyzwalacza. Zachowanie właściwości: - *wymagane*, jeśli `label_type_trigger` jest ustawione na ”custom”
label_type	integer	Typ etykiety elementu mapy. Możliwe wartości: 0 - etykieta; 1 - adres IP; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic.
label_type_host	integer	Typ etykiety dla elementów typu host. Możliwe wartości: 0 - etykieta; 1 - adres IP; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.
label_type_hostgroup	integer	Typ etykiety dla elementów grupy hostów. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.

Właściwość	Typ	Opis
label_type_image	integer	Typ etykiety dla elementów grupy hostów. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 4 - nic; 5 - niestandardowa.
label_type_map	integer	Typ etykiety dla elementów mapy. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.
label_type_trigger	integer	Typ etykiety dla elementów wyzwalacza. Możliwe wartości: 0 - etykieta; 2 - *(domyślnie)* nazwa elementu; 3 - tylko status; 4 - nic; 5 - niestandardowa.
markelements	integer	Czy podświetlać elementy mapy, których status ostatnio się zmienił. Możliwe wartości: 0 - *(domyślnie)* nie podświetlaj elementów; 1 - podświetlaj elementy.
severity_min	integer	Minimalny poziom ważności wyzwalaczy, które będą wyświetlane na mapie. Zobacz właściwość `severity` wyzwalacza, aby uzyskać listę obsługiwanych poziomów ważności wyzwalaczy.
show_element_label	integer	Jak domyślnie wyświetlać etykiety elementów. Możliwe wartości: 0 - zawsze wyświetlaj; 1 - *(domyślnie)* ukrywaj automatycznie.
show_link_label	integer	Jak domyślnie wyświetlać etykiety łączy. Możliwe wartości: 0 - zawsze wyświetlaj; 1 - *(domyślnie)* ukrywaj automatycznie.
show_unack	integer	Jak mają być wyświetlane problemy. Możliwe wartości: 0 - *(domyślnie)* wyświetlaj liczbę wszystkich problemów; 1 - wyświetlaj tylko liczbę niepotwierdzonych problemów; 2 - wyświetlaj osobno liczbę potwierdzonych i niepotwierdzonych problemów.
userid	ID	ID użytkownika będącego właścicielem mapy.
private	integer	Typ udostępniania mapy. Możliwe wartości: 0 - mapa publiczna; 1 - *(domyślnie)* mapa prywatna.
show_suppressed	integer	Czy wyciszone problemy są wyświetlane. Możliwe wartości: 0 - *(domyślnie)* ukrywaj wyciszone problemy; 1 - wyświetlaj wyciszone problemy.

Element mapy

Obiekt elementu mapy definiuje obiekt wyświetlany na mapie. Ma on następujące właściwości.

Właściwość	Typ	Opis
selementid	ID	ID elementu mapy. Zachowanie właściwości: - *tylko do odczytu*
elements	array	Obiekt danych elementu. Zachowanie właściwości: - *wymagane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
elementtype	integer	Typ elementu mapy. Możliwe wartości: 0 - host; 1 - mapa; 2 - wyzwalacz; 3 - grupa hostów; 4 - obraz. Zachowanie właściwości: - *wymagane*
iconid_off	ID	ID obrazu używanego do wyświetlania elementu w stanie domyślnym. Zachowanie właściwości: - *wymagane*
areatype	integer	Sposób wyświetlania oddzielnych hostów grupy hostów. Możliwe wartości: 0 - *(domyślnie)* element grupy hostów zajmie całą mapę; 1 - element grupy hostów będzie miał stały rozmiar.
elementsubtype	integer	Sposób wyświetlania elementu grupy hostów na mapie. Możliwe wartości: 0 - *(domyślnie)* wyświetl grupę hostów jako pojedynczy element; 1 - wyświetl każdy host w grupie oddzielnie.
evaltype	integer	Warunek filtrowania tagów elementu mapy, metoda obliczania. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
height	integer	Wysokość elementu grupy hostów o stałym rozmiarze w pikselach. Domyślnie: 200.
iconid_disabled	ID	ID obrazu używanego do wyświetlania wyłączonych elementów mapy. Zachowanie właściwości: - *obsługiwane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
iconid_maintenance	ID	ID obrazu używanego do wyświetlania elementów mapy w konserwacji. Zachowanie właściwości: - *obsługiwane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
iconid_on	ID	ID obrazu używanego do wyświetlania elementów mapy z problemami. Zachowanie właściwości: - *obsługiwane*, jeśli `elementtype` jest ustawione na ”host”, ”map”, ”trigger” lub ”host group”
label	string	Etykieta elementu.

Właściwość	Typ	Opis
label_location	integer	Położenie etykiety elementu mapy. Możliwe wartości: -1 - *(domyślnie)* położenie domyślne; 0 - dół; 1 - lewo; 2 - prawo; 3 - góra.
permission	integer	Typ poziomu uprawnień. Możliwe wartości: -1 - brak; 2 - tylko do odczytu; 3 - odczyt-zapis.
show_label	integer	Sposób wyświetlania etykiety elementu. Możliwe wartości: -1 - *(domyślnie)* domyślne ustawienie mapy; 0 - zawsze wyświetlaj; 1 - ukrywaj automatycznie.
sysmapid	ID	ID mapy, do której należy element. Zachowanie właściwości: - *tylko do odczytu*
urls	array	Adresy URL elementu mapy. Obiekt URL elementu mapy jest opisany szczegółowo poniżej.
use_iconmap	integer	Czy mapowanie ikon musi być używane dla elementów typu host. Możliwe wartości: 0 - nie używaj mapowania ikon; 1 - *(domyślnie)* używaj mapowania ikon.
viewtype	integer	Algorytm rozmieszczania elementu grupy hostów. Możliwe wartości: 0 - *(domyślnie)* siatka.
width	integer	Szerokość elementu grupy hostów o stałym rozmiarze w pikselach. Domyślnie: 200.
x	integer	Współrzędna X elementu w pikselach. Domyślnie: 0.
y	integer	Współrzędna Y elementu w pikselach. Domyślnie: 0.
zindex	integer	Wartość używana do porządkowania elementów mapy (z-index). Domyślnie: 0.

Element mapy Host

Obiekt elementu mapy Host definiuje jeden element hosta.

Właściwość	Typ	Opis
hostid	ID	ID hosta.

Element mapy Grupa hostów

Obiekt elementu mapy Grupa hostów definiuje jeden element grupy hostów.

Właściwość	Typ	Opis
groupid	ID	ID grupy hostów.

Obiekt elementu mapy Map

Obiekt elementu mapy Map definiuje jeden element mapy.

Właściwość	Type	Opis
sysmapid	ID	ID mapy.

Wyzwalacz elementu mapy

Obiekt wyzwalacza elementu mapy definiuje jeden lub więcej elementów wyzwalacza.

Właściwość	Type	Opis
triggerid	ID	ID wyzwalacza.

Tag elementu mapy

Obiekt Tag elementu mapy ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu elementu mapy. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek operatora tagu elementu mapy. Możliwe wartości: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
value	string	Wartość tagu elementu mapy.

URL elementu mapy

Obiekt URL elementu mapy definiuje klikalny link, który będzie dostępny dla określonego elementu mapy. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmapelementurlid	ID	ID URL elementu mapy. Zachowanie właściwości: - *tylko do odczytu*
name	string	Podpis linku. Zachowanie właściwości: - *wymagane*
url	string	URL linku. Zachowanie właściwości: - *wymagane*
selementid	ID	ID elementu mapy, do którego należy URL.

Łącze mapy

Obiekt łącza mapy definiuje połączenie między dwoma elementami mapy. Ma następujące właściwości.

Właściwość	Typ	Opis
linkid	ID	ID łącza mapy. Zachowanie właściwości: - *tylko do odczytu*
sysmapid	ID	ID mapy, do której należy łącze.
selementid1	ID	ID pierwszego elementu mapy połączonego z jednej strony. Zachowanie właściwości: - *wymagane*
selementid2	ID	ID pierwszego elementu mapy połączonego z drugiej strony. Zachowanie właściwości: - *wymagane*
drawtype	integer	Styl rysowania linii łącza. Możliwe wartości: 0 - *(domyślnie)* linia; 2 - linia pogrubiona; 3 - linia kropkowana; 4 - linia kreskowana.
color	string	Kolor linii jako szesnastkowy kod koloru. Domyślnie: 000000.
label	string	Etykieta łącza.
show_label	integer	Sposób wyświetlania etykiety łącza. Możliwe wartości: -1 - *(domyślnie)* domyślne ustawienie mapy; 0 - zawsze wyświetlaj; 1 - ukrywaj automatycznie.
indicator_type	integer	Wybór typu wskaźnika łącza. Możliwe wartości: 0 - *(domyślnie)* łącze statyczne; 1 - wyzwalacz; 2 - wartość pozycji.
linktriggers	array	Wyzwalacze łącza mapy używane jako wskaźniki stanu łącza. Obiekt wyzwalacza łącza mapy jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na ”trigger” (1).
itemid	ID	ID pozycji. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na ”item value” (2).
highlights	array	Wyróżnienia łącza mapy używane jako wskaźniki stanu łącza. Obiekt wskaźników łącza mapy jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na ”item value” (2).

Właściwość	Typ	Opis
thresholds	array	Progi łącza mapy używane jako wskaźniki stanu łącza. Obiekt wskaźników łącza mapy jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*, jeśli `indicator_type` jest ustawione na "item value" (2).
permission	integer	Typ poziomu uprawnień. Możliwe wartości: -1 - brak; 2 - tylko do odczytu; 3 - odczyt-zapis.

Wyzwalacz łącza mapy

Obiekt wyzwalacza łącza mapy definiuje wskaźnik stanu łącza mapy na podstawie stanu wyzwalacza. Ma następujące właściwości:

Właściwość	Typ	Opis
triggerid	ID	ID wyzwalacza używanego jako wskaźnik łącza. Zachowanie właściwości: - *wymagane*
color	string	Kolor wskaźnika jako szesnastkowy kod koloru. Domyślnie: `DD0000`.
drawtype	integer	Styl rysowania wskaźnika. Możliwe wartości: 0 - *(domyślnie)* linia; 2 - linia pogrubiona; 3 - linia kropkowana; 4 - linia przerywana.

Wskaźniki łączy mapy

Obiekt wskaźników łączy mapy definiuje wskaźnik stanu łącza mapy na podstawie wartości pozycji. Ma następujące właściwości:

Właściwość	Typ	Opis
pattern	string	Wyrażenie regularne do porównania. Dostępne tylko dla wyróżnień.
sortorder	integer	Używane do definiowania kolejności sortowania wyróżnień. Dostępne tylko dla wyróżnień.
threshold	string	Wartość liczbowa do porównania. Dostępne tylko dla progów.
drawtype	integer	Styl rysowania wskaźnika. Możliwe wartości: 0 - *(domyślnie)* linia; 2 - linia pogrubiona; 3 - linia kropkowana; 4 - linia kreskowana.
color	string	Kolor wskaźnika jako szesnastkowy kod koloru. Domyślnie: `DD0000`. Zachowanie właściwości: - *wymagane*

URL mapy

Obiekt URL mapy definiuje klikalny link, który będzie dostępny dla wszystkich elementów określonego typu na mapie. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmapurlid	ID	ID URL mapy. Zachowanie właściwości: - *tylko do odczytu*
name	string	Podpis linku. Zachowanie właściwości: - *wymagane*
url	string	URL linku. Zachowanie właściwości: - *wymagane*
elementtype	integer	Typ elementu mapy, dla którego URL będzie dostępny. Listę obsługiwanych typów można znaleźć we właściwości `type` elementu mapy. Domyślnie: 0.
sysmapid	ID	ID mapy, do której należy URL.

Użytkownik mapy

Lista uprawnień do mapy na podstawie użytkowników. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmapuserid	ID	ID użytkownika mapy. Zachowanie właściwości: - *tylko do odczytu*
userid	ID	ID użytkownika. Zachowanie właściwości: - *wymagane*
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Zachowanie właściwości: - *wymagane*

Grupa użytkowników mapy

Lista uprawnień do mapy opartych na grupach użytkowników. Ma następujące właściwości:

Właściwość	Type	Opis
sysmapusrgrpid	ID	ID grupy użytkowników mapy. Property behavior: - *tylko do odczytu*
usrgrpid	ID	ID grupy użytkowników. Property behavior: - *wymagane*

Właściwość	Type	Opis
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Property behavior: - *wymagane*

Kształty mapy

Obiekt kształtu mapy definiuje kształt geometryczny (z tekstem lub bez) wyświetlany na mapie. Ma następujące właściwości:

Właściwość	Type	Opis
sysmap_shapeid	ID	ID elementu kształtu mapy. Zachowanie właściwości: - *tylko do odczytu*
type	integer	Typ elementu kształtu mapy. Możliwe wartości: 0 - prostokąt; 1 - elipsa. Właściwość jest wymagana podczas tworzenia nowych kształtów. Zachowanie właściwości: - *wymagane*
x	integer	Współrzędne X kształtu w pikselach. Domyślnie: 0.
y	integer	Współrzędne Y kształtu w pikselach. Domyślnie: 0.
width	integer	Szerokość kształtu w pikselach. Domyślnie: 200.
height	integer	Wysokość kształtu w pikselach. Domyślnie: 200.
text	string	Tekst kształtu.
font	integer	Czcionka tekstu wewnątrz kształtu. Możliwe wartości: 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 Domyślnie: 9.

Właściwość	Type	Opis
font_size	integer	Rozmiar czcionki w pikselach. Domyślnie: 11.
font_color	string	Kolor czcionki. Domyślnie: 000000.
text_halign	integer	Wyrównanie poziome tekstu. Możliwe wartości: 0 - środek; 1 - do lewej; 2 - do prawej. Domyślnie: 0.
text_valign	integer	Wyrównanie pionowe tekstu. Możliwe wartości: 0 - środek; 1 - góra; 2 - dół. Domyślnie: 0.
border_type	integer	Typ obramowania. Możliwe wartości: 0 - brak; 1 - ————; 2 - ┅; 3 - – – –. Domyślnie: 0.
border_width	integer	Szerokość obramowania w pikselach. Domyślnie: 0.
border_color	string	Kolor obramowania. Domyślnie: 000000.
background_color	string	Kolor tła (kolor wypełnienia). Domyślnie: (puste).
zindex	integer	Wartość używana do porządkowania wszystkich kształtów i linii (z-index). Domyślnie: 0.

Linie mapy

Obiekt linii mapy definiuje linię wyświetlaną na mapie. Ma następujące właściwości:

Właściwość	Typ	Opis
sysmap_shapeid	ID	ID elementu kształtu mapy. Zachowanie właściwości: *- tylko do odczytu*
x1	integer	Współrzędne X punktu 1 linii w pikselach. Domyślnie: 0.
y1	integer	Współrzędne Y punktu 1 linii w pikselach. Domyślnie: 0.

Właściwość	Typ	Opis
x2	integer	Współrzędne X punktu 2 linii w pikselach. Domyślnie: 200.
y2	integer	Współrzędne Y punktu 2 linii w pikselach. Domyślnie: 200.
line_type	integer	Typ linii. Możliwe wartości: 0 - brak; 1 - ——————; 2 - ┅; 3 - – – –. Domyślnie: 0.
line_width	integer	Szerokość linii w pikselach. Domyślnie: 0.
line_color	string	Kolor linii. Domyślnie: 000000.
zindex	integer	Wartość używana do porządkowania wszystkich kształtów i linii (z-index). Domyślnie: 0.

map.create

Opis

`object map.create(object/array maps)`

Ta metoda umożliwia tworzenie nowych map.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Mapy do utworzenia.

Oprócz standardowych właściwości mapy, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
links	array	Łącza mapy do utworzenia na mapie.
selements	array	Elementy mapy do utworzenia na mapie.
urls	array	Adresy URL mapy do utworzenia na mapie.
users	array	Udostępnienia użytkownikowi mapy do utworzenia na mapie.
userGroups	array	Udostępnienia grupie użytkowników mapy do utworzenia na mapie.
shapes	array	Kształty mapy do utworzenia na mapie.
lines	array	Linie mapy do utworzenia na mapie.

Note:
Aby utworzyć łącza mapy, należy ustawić `selementid` elementu mapy na dowolną wartość, a następnie użyć tej wartości do odwołania się do tego elementu we właściwościach łączy `selementid1` lub `selementid2`. Po utworzeniu elementu wartość ta zostanie zastąpiona poprawnym identyfikatorem wygenerowanym przez Zabbix. Zobacz przykład.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych map we właściwości sysmapids. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych map.

Przykłady

Utwórz pustą mapę

Utwórz mapę bez elementów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Map",
        "width": 600,
        "height": 600
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Utwórz mapę hostów

Utwórz mapę z dwoma elementami hostów i łączem między nimi. Zwróć uwagę na użycie tymczasowych wartości „selementid1” i „selementid2” w obiekcie łącza mapy do odwoływania się do elementów mapy.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Utwórz mapę wyzwalaczy

Utwórz mapę z elementem wyzwalacza, który zawiera dwa wyzwalacze.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.create",
    "params": {
        "name": "Mapa wyzwalaczy",
        "width": 600,
        "height": 600,
        "selements": [
            {
                "elements": [
                    {"triggerid": "12345"},
                    {"triggerid": "67890"}
                ],
                "elementtype": 2,
                "iconid_off": "2"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "10"
        ]
    },
    "id": 1
}
```

Udostępnianie mapy

Utwórz mapę z dwoma typami udostępniania (użytkownik i grupa użytkowników).

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Kształty mapy

Utwórz mapę z tytułem będącym nazwą mapy.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
```

```
            "10"
        ]
    },
    "id": 1
}
```

Linie mapy

Utwórz linię mapy.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "11"
        ]
    },
    "id": 1
}
```

Zobacz także

- Element mapy
- Łącze mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształt mapy
- Linia mapy

Źródło

CMap::create() w *ui/include/classes/api/services/CMap.php*.

map.delete

Opis

`object map.delete(array mapIds)`

Ta metoda umożliwia usuwanie map.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(array)` ID map do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych map we właściwości `sysmapids`.

Przykłady

Usuń wiele map

Usuń dwie mapy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.delete",
    "params": [
        "12",
        "34"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "12",
            "34"
        ]
    },
    "id": 1
}
```

Źródło

CMap::delete() w *ui/include/classes/api/services/CMap.php*.

map.get

Opis

`integer/array map.get(object parameters)`

Ta metoda umożliwia pobieranie map zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
sysmapids	ID/array	Zwraca tylko mapy o podanych identyfikatorach.
userids	ID/array	Zwraca tylko mapy należące do użytkowników o podanych identyfikatorach.

Parametr	Typ	Opis
expandUrls	flag	Dodaje globalne adresy URL map do odpowiednich elementów mapy i rozwija makra we wszystkich adresach URL elementów mapy.
selectIconMap	query	Zwraca właściwość `iconmap` z mapą ikon używaną na mapie.
selectLinks	query	Zwraca właściwość `links` z połączeniami mapy między elementami.
selectSelements	query	Zwraca właściwość `selements` z elementami mapy.
selectUrls	query	Zwraca właściwość `urls` z adresami URL mapy.
selectUsers	query	Zwraca właściwość `users` z użytkownikami, którym mapa jest udostępniona.
selectUserGroups	query	Zwraca właściwość `userGroups` z grupami użytkowników, którym mapa jest udostępniona.
selectShapes	query	Zwraca właściwość `shapes` z kształtami mapy.
selectLines	query	Zwraca właściwość `lines` z liniami mapy.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `name`, `width`, `height`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie mapy

Pobierz wszystkie dane o mapie „3”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Mapa ikon
- Element mapy
- Łącze mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształty mapy
- Linie mapy

Źródło

CMap::get() w *ui/include/classes/api/services/CMap.php*.

map.update

Opis

`object map.update(object/array maps)`

Ta metoda umożliwia aktualizację istniejących map.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Parametry

`(object/array)` Właściwości mapy do zaktualizowania.

Właściwość `mapid` musi być zdefiniowana dla każdej mapy, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości mapy metoda akceptuje następujące parametry.

Parametr	Type	Opis
links	array	Łącza mapy zastępujące istniejące łącza.
selements	array	Elementy mapy zastępujące istniejące elementy.
urls	array	Adresy URL mapy zastępujące istniejące adresy URL.
users	array	Udostępnienia użytkownikowi mapy zastępujące istniejące elementy.
userGroups	array	Udostępnienia grupie użytkowników mapy zastępujące istniejące elementy.
shapes	array	Kształty mapy zastępujące istniejące kształty.
lines	array	Linie mapy zastępujące istniejące linie.

> **Note:**
> Aby utworzyć łącza mapy między nowymi elementami mapy, należy ustawić `selementid` elementu na dowolną wartość, a następnie użyć tej wartości do odwołania się do tego elementu we właściwościach łączy `selementid1` lub `selementid2`. Po utworzeniu elementu wartość ta zostanie zastąpiona poprawnym ID wygenerowanym przez Zabbix. Zobacz przykład dla map.create.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych map we właściwości `sysmapids`.

Przykłady

Zmień rozmiar mapy

Zmień rozmiar mapy na 1200x1200 pikseli.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "8",
        "width": 1200,
        "height": 1200
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}
```

Zmień właściciela mapy

Dostępne tylko dla administratorów i super administratorów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "9",
        "userid": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 1
}
```

Zobacz także

- Element mapy
- Łącze mapy
- URL mapy
- Użytkownik mapy
- Grupa użytkowników mapy
- Kształty mapy
- Linie mapy

Źródło

CMap::update() w *ui/include/classes/api/services/CMap.php*.

Mapa ikon

Ta klasa jest przeznaczona do pracy z mapami ikon.

Odwołania do obiektów:

- Mapa ikon
- Mapowanie ikon

Dostępne metody:

- iconmap.create - utwórz nowe mapy ikon
- iconmap.delete - usuń mapy ikon
- iconmap.get - pobierz mapy ikon
- iconmap.update - zaktualizuj mapy ikon

Obiekt mapy ikon

Następujące obiekty są bezpośrednio powiązane z API `iconmap`.

Mapa ikon

Obiekt mapy ikon ma następujące właściwości.

Właściwość	Typ	Opis
iconmapid	ID	ID mapy ikon. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
default_iconid	ID	ID domyślnej ikony. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa mapy ikon. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Mapowanie ikon

Obiekt mapowania ikon definiuje konkretną ikonę, która ma być używana dla hostów o określonej wartości pola inwentarza. Ma następujące właściwości.

Właściwość	Typ	Opis
iconid	ID	ID ikony używanej przez mapowanie ikon. Zachowanie właściwości: - *wymagane*
expression	string	Wyrażenie służące do dopasowania do pola inwentarza. Zachowanie właściwości: - *wymagane*
inventory_link	integer	ID pola inwentarza hosta. Listę obsługiwanych pól inwentarza można znaleźć w obiekcie inwentarza hosta. Zachowanie właściwości: - *wymagane*
sortorder	integer	Pozycja mapowania ikon na mapie ikon. Zachowanie właściwości: - *tylko do odczytu*

iconmap.create

Opis

`object iconmap.create(object/array iconMaps)`

Ta metoda umożliwia tworzenie nowych map ikon.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników

Parametry

`(object/array)` Mapy ikon do utworzenia.

Oprócz standardowych właściwości map ikon, metoda akceptuje następujące parametry.

Parametr	Type	Opis
mappings	array	Mapowania ikon, które mają zostać utworzone dla mapy ikon. Zachowanie parametrów: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych map ikon we właściwości `iconmapids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych map ikon.

Przykłady

Utwórz mapę ikon

Utwórz mapę ikon, aby wyświetlać hosty różnych typów.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2"
        ]
```

```
    },
    "id": 1
}
```

Zobacz także

- Mapowanie ikon

Źródło

CIconMap::create() w *ui/include/classes/api/services/CIconMap.php*.

iconmap.delete

Opis

`object iconmap.delete(array iconMapIds)`

Ta metoda umożliwia usuwanie map ikon.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID map ikon do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych map ikon we właściwości `iconmapids`.

Przykłady

Usuń wiele map ikon

Usuń dwie mapy ikon.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.delete",
    "params": [
        "2",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "2",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CIconMap::delete() w *ui/include/classes/api/services/CIconMap.php*.

iconmap.get

Opis

`integer/array iconmap.get(object parameters)`

Metoda umożliwia pobranie map ikon zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
iconmapids	ID/array	Zwraca tylko mapy ikon o podanych identyfikatorach.
sysmapids	ID/array	Zwraca tylko mapy ikon używane w podanych mapach.
selectMappings	query	Zwraca właściwość `mappings` z używanymi mapowaniami ikon.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `iconmapid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie mapy ikon

Pobierz wszystkie dane o mapie ikon „3”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.get",
    "params": {
        "iconmapids": "3",
        "output": "extend",
        "selectMappings": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Mapowanie ikon

Źródło

CIconMap::get() w *ui/include/classes/api/services/CIconMap.php*.

iconmap.update

Opis

`object iconmap.update(object/array iconMaps)`

Ta metoda umożliwia aktualizację istniejących map ikon.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości mapy ikon do zaktualizowania.

Właściwość `iconmapid` musi być zdefiniowana dla każdej mapy ikon, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości mapy ikon metoda akceptuje następujące parametry.

Parametr	Type	Opis
mappings	array	Mapowania ikon zastępujące istniejące mapowania ikon.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych map ikon we właściwości `iconmapids`.

Przykłady

Zmień nazwę mapy ikon

Zmień nazwę mapy ikon na „OS icons”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "iconmap.update",
    "params": {
        "iconmapid": "1",
        "name": "OS icons"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "iconmapids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Mapowanie ikon

Źródło

CIconMap::update() w *ui/include/classes/api/services/CIconMap.php*.

Mapa wartości

Ta klasa jest przeznaczona do pracy z mapami wartości.

Odwołania do obiektów:

- Mapa wartości
  - Mapowania wartości

Dostępne metody:

- valuemap.create - tworzenie nowych map wartości
- valuemap.delete - usuwanie map wartości
- valuemap.get - pobieranie map wartości
- valuemap.update - aktualizowanie map wartości

Obiekt mapy wartości

Następujące obiekty są bezpośrednio powiązane z API `valuemap`.

Mapa wartości

Obiekt mapy wartości ma następujące właściwości.

Właściwość	Typ	Opis
valuemapid	ID	ID mapy wartości. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Właściwość	Typ	Opis
hostid	ID	ID hosta lub szablonu, do którego należy mapa wartości. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
name	string	Nazwa mapy wartości. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
mappings	array	Mapowania wartości dla bieżącej mapy wartości. Obiekt mapowania jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych map wartości z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli mapa wartości należy do szablonu

Mapowania wartości

Obiekt mapowań wartości definiuje mapowania wartości mapy wartości. Ma następujące właściwości.

Właściwość	Type	Opis
type	integer	Typ dopasowania mapowania. Możliwe wartości: 0 - *(domyślnie)* mapowanie zostanie zastosowane, jeśli wartość jest równa; 1 - mapowanie zostanie zastosowane, jeśli wartość jest większa lub równa[1]; 2 - mapowanie zostanie zastosowane, jeśli wartość jest mniejsza lub równa[1]; 3 - mapowanie zostanie zastosowane, jeśli wartość mieści się w zakresie (zakresy są domknięte; można zdefiniować wiele zakresów oddzielonych przecinkiem)[1]; 4 - mapowanie zostanie zastosowane, jeśli wartość pasuje do wyrażenia regularnego[2]; 5 - jeśli nie zostaną znalezione żadne dopasowania, mapowanie nie zostanie zastosowane i zostanie użyta wartość domyślna. Jeśli `type` jest ustawione na ”0”, ”1”, ”2”, ”3”, ”4”, wówczas `value` nie może być puste. Jeśli `type` jest ustawione na ”5”, wówczas `value` musi być puste.
value	string	Wartość oryginalna. Zachowanie właściwości: - *wymagane* jeśli `type` jest ustawione na ”1”, ”2”, ”3”, ”4” - *obsługiwane* jeśli `type` jest ustawione na ”5”
newvalue	string	Wartość, na którą mapowana jest wartość oryginalna. Zachowanie właściwości: - *wymagane*

[1] obsługiwane tylko dla pozycji mających typ wartości ”numeric unsigned”, ”numeric float”.
 [2] obsługiwane tylko dla pozycji mających typ wartości ”character”.

valuemap.create

Opis

`object valuemap.create(object/array valuemaps)`

Ta metoda umożliwia tworzenie nowych map wartości.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Mapowania wartości do utworzenia.

Metoda akceptuje mapowania wartości ze standardowymi właściwościami mapowania wartości.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory właściwości `valuemapids` utworzonych map wartości. Kolejność zwróconych identyfikatorów jest zgodna z kolejnością przekazywanych map wartości.

Przykłady

Tworzenie mapowania wartości

Utwórz jedno mapowanie wartości z dwoma odwzorowaniami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.create",
    "params": {
        "hostid": "50009",
        "name": "Stan usługi",
        "mappings": [
            {
                "type": "1",
                "value": "1",
                "newvalue": "Działa"
            },
            {
                "type": "5",
                "newvalue": "Niedostępna"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CValueMap::create() in *ui/include/classes/api/services/CValueMap.php*.

valuemap.delete

Opis

`object valuemap.delete(array valuemapids)`

Ta metoda umożliwia usuwanie map wartości.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` Identyfikatory map wartości do usunięcia.

Zwracane wartości

`(array)` Zwraca obiekt zawierający identyfikatory usuniętych map wartości we właściwości `valuemapids`.

Przykłady

Usuwanie wielu map wartości

Usuń dwie mapy wartości.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "valuemap.delete",
"params": [
"1",
"2"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"valuemapids": [
"1",
"2"
]
},
"id": 1
}
```

Źródło

CValueMap::delete() in *ui/include/classes/api/services/CValueMap.php*.

valuemap.get

Opis

`integer/array valuemap.get(object parameters)`

Metoda umożliwia pobieranie map wartości zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

(object) Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
valuemapids	ID/array	Zwracaj tylko mapowania wartości o podanych identyfikatorach.
selectMappings	query	Zwracaj mapowania wartości dla bieżącej mapy wartości we właściwości `mappings`. Obsługuje `count`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `valuemapid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwróć wartości

(integer/array) Zwraca albo:

- tablicę obiektów;
- ilość pobranych obiektów, jeśli parametr `countOutput` został użyty.

Przykłady

Pobieranie map wartości

Pobierz wszystkie skonfigurowane mapy wartości.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobierz jedną mapę wartości wraz z jej mapowaniami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.get",
    "params": {
        "output": "extend",
        "selectMappings": "extend",
        "valuemapids": ["4"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CValueMap::get() in *ui/include/classes/api/services/CValueMap.php*.

valuemap.update

Opis

`object valuemap.update(object/array valuemaps)`

Ta metoda umożliwia aktualizację istniejących map wartości.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości mapy wartości, które mają zostać zaktualizowane.

Właściwość `valuemapid` musi być zdefiniowana dla każdej mapy wartości, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory ID zaktualizowanych wartości map we własności `valuemapids`.

Przykłady

Zmiana nazwy mapy wartości

Zmień nazwę mapy wartości na „Device status”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "name": "Device status"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Zmienianie mapowań dla jednej mapy wartości.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
```

```
        "valuemapid": "2",
        "mappings": [
            {
                "type": "0",
                "value": "0",
                "newvalue": "Online"
            },
            {
                "type": "0",
                "value": "1",
                "newvalue": "Offline"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "valuemapids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CValueMap::update() w *ui/include/classes/api/services/CValueMap.php*.

Metryka APM

Ta klasa jest przeznaczona do pracy z metrykami zbieranymi na potrzeby monitorowania wydajności aplikacji (APM).

Odwołania do obiektów:

- Metryka APM
- Przykład

Dostępne metody:

- apmmetric.get - pobieranie punktów danych metryki

Obiekt metryki APM

Poniższe obiekty są bezpośrednio powiązane z API `apmmetric`.

Note:
Każda właściwość, z wyjątkiem `type`, odpowiada kolumnie w tabelach metryk (`otel_metrics_gauge`, `otel_metrics_sum`, `otel_metrics_histogram` i `otel_metrics_exponential_histogram`) źródła danych telemetrycznych. Większość właściwości odpowiada kolumnie o tej samej nazwie (np. właściwość `metric_name` odpowiada kolumnie `MetricName`). Jeśli nazwy się różnią, kolumny źródła danych są określone w poniższych tabelach. Listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych.

Metryka APM

Obiekt metryki APM reprezentuje punkt danych metryki (wartość lub zagregowane wartości pomiaru w określonym czasie).

Obiekt metryki APM ma następujące właściwości:

Właściwość	Typ	Opis
type	integer	Typ punktu danych. Możliwe wartości: 0 - Gauge; 1 - Sum; 2 - Histogram; 3 - Histogram wykładniczy. Typ jest określany przez tabelę przechowującą punkt danych: `otel_metrics_gauge`, `otel_metrics_sum`, `otel_metrics_histogram` lub `otel_metrics_exponential_histogram`. Zachowanie właściwości: - *tylko do odczytu*
resource_attributes	object	Atrybuty zasobu punktu danych w postaci klucz-wartość (np. `host.name: eabf8c2a5ddc`). Zachowanie właściwości: - *tylko do odczytu*
resource_schema_url	string	Adres URL, jeśli istnieje, schematu zasobu OpenTelemetry dla atrybutów zasobu. Zachowanie właściwości: - *tylko do odczytu*
scope_name	string	Nazwa zakresu instrumentacji, który wygenerował punkt danych. Zachowanie właściwości: - *tylko do odczytu*
scope_version	string	Wersja zakresu instrumentacji, jeśli istnieje. Zachowanie właściwości: - *tylko do odczytu*
scope_attributes	object	Atrybuty zakresu instrumentacji w postaci klucz-wartość. Zachowanie właściwości: - *tylko do odczytu*
scope_schema_url	string	Adres URL, jeśli istnieje, schematu zakresu instrumentacji OpenTelemetry i jego atrybutów. Zachowanie właściwości: - *tylko do odczytu*
service_name	string	Nazwa usługi, która wygenerowała punkt danych (np. `zabbix-frontend`), pobrana z atrybutu zasobu OpenTelemetry `service.name`. Zachowanie właściwości: - *tylko do odczytu*
metric_name	string	Nazwa metryki, do której należy punkt danych (np. `http.server.request.duration`). Zachowanie właściwości: - *tylko do odczytu*
metric_description	string	Opis metryki, jeśli istnieje. Zachowanie właściwości: - *tylko do odczytu*
metric_unit	string	Jednostka wartości metryki, jeśli istnieje (np. `s` lub `By`; zobacz UCUM). Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
attributes	object	Atrybuty punktu danych w postaci klucz-wartość (np. `http.request.method: POST`). Zachowanie właściwości: - *tylko do odczytu*
start_time_unix	integer	Czas uniksowy (w nanosekundach), w którym rozpoczął się okres pomiaru punktu danych. Zachowanie właściwości: - *tylko do odczytu*
time_unix	integer	Czas uniksowy (w nanosekundach) punktu danych. W przypadku punktów danych typu *Sum*, *Histogram* i *Histogram wykładniczy* jest to koniec okresu pomiaru. Zachowanie właściwości: - *tylko do odczytu*
value	float	Wartość punktu danych. Dotyczy punktów danych typu *Gauge* i *Sum*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*
flags	integer	Maska bitowa flag punktu danych. Jeśli ustawiony jest najmniej znaczący bit (wartość 1), dla punktu danych nie zarejestrowano żadnej wartości. Zachowanie właściwości: - *tylko do odczytu*
exemplars	array	Tablica obiektów exemplar, czyli pojedynczych pomiarów zachowanych jako przykłady wraz z identyfikatorem śladu i rozpiętością, podczas których zostały zarejestrowane. Zachowanie właściwości: - *tylko do odczytu*
aggregation_temporality	integer	Sposób agregowania wartości punktu danych w czasie. Dotyczy punktów danych typu *Sum*, *Histogram* i *Histogram wykładniczy*. Możliwe wartości: 0 - *(domyślnie)* Nieokreślony; 1 - Delta (agregowane w okresie od poprzedniego punktu danych); 2 - Skumulowane (agregowane w okresie od `start_time_unix`). Zachowanie właściwości: - *tylko do odczytu*
is_monotonic	boolean	Określa, czy suma może z czasem tylko rosnąć lub pozostać bez zmian. Dotyczy punktów danych typu *Sum*. Wartość domyślna: `false`. Zachowanie właściwości: - *tylko do odczytu*
count	integer	Liczba pomiarów w punkcie danych. Dotyczy punktów danych typu *Histogram* i *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
sum	float	Suma wszystkich pomiarów w punkcie danych. Dotyczy punktów danych typu *Histogram* i *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*
bucket_counts	array	Tablica liczb całkowitych określających liczbę pomiarów w każdym przedziale histogramu. Przedziały są definiowane przez `explicit_bounds`. Dotyczy punktów danych typu *Histogram*. Wartość domyślna: `[]`. Zachowanie właściwości: - *tylko do odczytu*
explicit_bounds	array	Tablica liczb zmiennoprzecinkowych określających granice między przedziałami histogramu. Dotyczy punktów danych typu *Histogram*. Wartość domyślna: `[]`. Zachowanie właściwości: - *tylko do odczytu*
min	float	Najmniejszy z pomiarów w punkcie danych. Dotyczy punktów danych typu *Histogram* i *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*
max	float	Największy z pomiarów w punkcie danych. Dotyczy punktów danych typu *Histogram* i *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*
scale	integer	Skala histogramu wykładniczego, która określa granice przedziałów i rozdzielczość. Wyższa skala zapewnia węższe przedziały i większą rozdzielczość, natomiast niższa skala zapewnia szersze przedziały i mniejszą rozdzielczość. Dotyczy punktów danych typu *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*
zero_count	integer	Liczba pomiarów w zerowym przedziale histogramu wykładniczego. Dotyczy punktów danych typu *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
positive_bucket_counts	array	Tablica liczb całkowitych określających liczbę dodatnich pomiarów w każdym przedziale histogramu wykładniczego. Granice przedziałów są określane przez `scale`; indeks pierwszego przedziału jest ustawiany w `positive_offset`. Dotyczy punktów danych typu *Histogram wykładniczy*. Wartość domyślna: `[]`. Zachowanie właściwości: - *tylko do odczytu*
positive_offset	integer	Indeks pierwszego przedziału w `positive_bucket_counts`. Dotyczy punktów danych typu *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*
negative_bucket_counts	array	Tablica liczb całkowitych określających liczbę ujemnych pomiarów w każdym przedziale histogramu wykładniczego. Granice przedziałów są określane przez `scale`; indeks pierwszego przedziału jest ustawiany w `negative_offset`. Dotyczy punktów danych typu *Histogram wykładniczy*. Wartość domyślna: `[]`. Zachowanie właściwości: - *tylko do odczytu*
negative_offset	integer	Indeks pierwszego przedziału w `negative_bucket_counts`. Dotyczy punktów danych typu *Histogram wykładniczy*. Wartość domyślna: 0. Zachowanie właściwości: - *tylko do odczytu*

Obiekt Exemplar

Obiekt exemplar ma następujące właściwości:

Property	Type	Description
traceid	string	Identyfikator (jeśli istnieje) śladu, podczas którego zarejestrowano pomiar. Kolumna źródła danych: `Exemplars.TraceId`.
spanid	string	Identyfikator (jeśli istnieje) zakresu, podczas którego zarejestrowano pomiar. Kolumna źródła danych: `Exemplars.SpanId`.
value	float	Wartość pomiaru. Kolumna źródła danych: `Exemplars.Value`.
time_unix	integer	Czas uniksowy (w nanosekundach), w którym zarejestrowano pomiar. Kolumna źródła danych: `Exemplars.TimeUnix`.
filtered_attributes	object	Atrybuty pomiaru w postaci klucz-wartość, które nie są uwzględnione w `attributes` punktu danych. Kolumna źródła danych: `Exemplars.FilteredAttributes`.

apmmetric.get

Opis

`integer/array apmmetric.get(object parameters)`

Metoda umożliwia pobieranie punktów danych metryki zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
time_from	timestamp	Zwraca tylko punkty danych, których czas jest równy podanemu czasowi lub późniejszy. Zachowanie parametru: - *wymagany*
time_till	timestamp	Zwraca tylko punkty danych, których czas jest wcześniejszy niż podany czas. Zachowanie parametru: - *wymagany*
types	integer/array	Zwraca tylko punkty danych podanych typów. Lista obsługiwanych typów metryk znajduje się na stronie obiektu metryki APM.
with_flags_on	integer	Zwraca tylko punkty danych, dla których ustawione są wszystkie podane flagi. Aby określić kilka flag, użyj sumy ich wartości.
with_flags_off	integer	Zwraca tylko punkty danych, dla których nie jest ustawiona żadna z podanych flag. Aby określić kilka flag, użyj sumy ich wartości.
resource_attributes	object/array	Zwraca tylko punkty danych z podanymi atrybutami zasobu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie punkty danych. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
resource_attributes_evaltype	integer	Metoda oceny warunków w `resource_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
scope_attributes	object/array	Zwraca tylko punkty danych z podanymi atrybutami zakresu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie punkty danych. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.

Parametr	Typ	Opis
scope_attributes_evaltype	integer	Metoda oceny warunków w `scope_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
attributes	object/array	Zwraca tylko punkty danych z podanymi atrybutami. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie punkty danych. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
attributes_evaltype	integer	Metoda oceny warunków w `attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami pojedyncza wartość lub tablica wartości, z którymi należy porównać wyniki. Obsługiwane właściwości: `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `service_name`, `metric_name`, `metric_unit`, `flags`.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami ciągi znaków, których należy szukać. Jeśli nie podano dodatkowych opcji, wyszukiwanie będzie wykonywane za pomocą `LIKE "%…%"`. Obsługiwane właściwości: `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `service_name`, `metric_name`, `metric_description`, `metric_unit`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `type`, `resource_schema_url`, `scope_schema_url`, `scope_name`, `scope_version`, `service_name`, `metric_name`, `metric_unit`, `start_time_unix`, `time_unix`.
offset	integer	Liczba rekordów pomijanych przed zwróceniem wyniku.
countOutput	boolean	Parametry te opisano w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Wartości zwracane

(integer/array) Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie punktów danych histogramu

Pobierz najnowszy punkt danych metryki histogramu `http.server.request.duration` usługi `zabbix-frontend`.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Ślad APM
- Zakres APM
- Dziennik APM

Źródło

CApmMetric::get() w *ui/include/classes/api/services/CApmMetric.php*.

MFA

Ta klasa jest przeznaczona do pracy z metodami MFA (uwierzytelnianie wieloskładnikowe).

Odwołania do obiektów:

- MFA

Dostępne metody:

- mfa.create - tworzenie nowych metod MFA
- mfa.delete - usuwanie metod MFA
- mfa.get - pobieranie metod MFA
- mfa.update - aktualizowanie metod MFA

Obiekt MFA (Multi-Factor Authentication)

Następujące obiekty są bezpośrednio powiązane z API `mfa`.

MFA

Obiekt MFA (uwierzytelnianie wieloskładnikowe) ma następujące właściwości.

Właściwość	Typ	Opis
mfaid	ID	ID metody MFA. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
type	integer	Typ metody MFA. Możliwe wartości: 1 - TOTP (jednorazowe hasła oparte na czasie); 2 - Duo Universal Prompt.
name	string	Unikalna nazwa metody MFA. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
hash_function	integer	Typ funkcji skrótu używanej do generowania kodów TOTP. Możliwe wartości: 1 - SHA-1; 2 - SHA-256; 3 - SHA-512. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „TOTP”
code_length	integer	Długość kodu weryfikacyjnego. Możliwe wartości: 6 - 6 cyfr; 8 - 8 cyfr. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „TOTP”
api_hostname	string	Nazwa hosta API dostarczona przez usługę uwierzytelniania Duo. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Duo Universal Prompt”

Właściwość	Typ	Opis
clientid	string	ID klienta dostarczone przez usługę uwierzytelniania Duo. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Duo Universal Prompt”
client_secret	string	Tajny klucz klienta dostarczony przez usługę uwierzytelniania Duo. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `type` jest ustawione na „Duo Universal Prompt”

mfa.create

Opis

`object mfa.create(object/array MFA methods)`

Ta metoda umożliwia tworzenie nowych metod MFA.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Metody MFA do utworzenia.

Ta metoda akceptuje metody MFA ze standardowymi właściwościami metod MFA.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych metod MFA we właściwości `mfaids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych pozycji.

Przykłady

Tworzenie metod MFA

Utwórz metodę MFA „Zabbix TOTP” wykorzystującą jednorazowe hasła oparte na czasie (TOTP), z funkcją skrótu do generowania kodów TOTP ustawioną na SHA-1 oraz długością kodu weryfikacyjnego ustawioną na 6 cyfr.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.create",
    "params": {
        "type": 1,
        "name": "Zabbix TOTP",
        "hash_function": 1,
        "code_length": 6
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CMfa::create() w *ui/include/classes/api/services/CMfa.php*.

mfa.delete

Opis

`object mfa.delete(array mfaids)`

Ta metoda umożliwia usuwanie metod MFA.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID metod MFA do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych metod MFA we właściwości `mfaids`.

Przykłady

Usuwanie metod MFA

Usuń metodę MFA.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.delete",
    "params": [
        "2"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CMfa::delete() w *ui/include/classes/api/services/CMfa.php*.

mfa.get

Opis

`integer/array mfa.get(object parameters)`

Metoda umożliwia pobranie metod MFA zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
mfaids	ID/array	Zwraca tylko metody MFA o podanych identyfikatorach.
selectUsrgrps	query	Zwraca właściwość `usrgrps` z grupami użytkowników powiązanymi z metodami MFA. Obsługuje `count`.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Obsługiwane właściwości: `mfaid` - identyfikator metody MFA; `type` - typ metody MFA.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `name`.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługiwane właściwości: `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Wyszukiwanie metod MFA po nazwie

Pobierz wszystkie metody MFA, które mają w nazwie „Zabbix”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.get",
    "params": {
        "output": "extend",
        "search": {
            "name": "Zabbix"
        }
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CMfa::get() w *ui/include/classes/api/services/CMfa.php*.

mfa.update

Opis

`object mfa.update(object/array MFA methods)`

Ta metoda umożliwia aktualizację istniejących metod MFA.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości metody MFA do zaktualizowania.

Właściwość `mfaid` musi być zdefiniowana dla każdej pozycji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje metody MFA ze standardowymi właściwościami metody MFA.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych metod MFA we właściwości `mfaids`.

Przykłady

Aktualizowanie właściwości metody

Zaktualizuj funkcję skrótu używaną do generowania kodów TOTP oraz długość kodu weryfikacyjnego dla metody MFA „Zabbix TOTP” wykorzystującej jednorazowe hasła oparte na czasie (TOTP).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mfa.update",
    "params": {
        "mfaid": "1",
        "hash_function": 3,
        "code_length": 8
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mfaids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CMfa::update() w *ui/include/classes/api/services/CMfa.php*.

Moduły

Ta klasa została zaprojektowana do współpracy z modułami frontend.

Odwołania do obiektów:

- Module

Dostępne metody:

- module.create - instalowanie nowych modułów
- module.delete - odinstalowywanie modułów
- module.get - pobieranie modułów
- module.update - aktualizowanie modułów

Obiekt modułu

Następujące obiekty są bezpośrednio powiązane z API `module`.

Moduł

Obiekt modułu ma następujące właściwości.

Property	Type	Description
moduleid	ID	ID modułu zapisane w bazie danych. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
id	string	Unikalne ID modułu zdefiniowane przez dewelopera w pliku `manifest.json` modułu. Możliwe wartości dla wbudowanych modułów: zobacz opis właściwości "type" w Dashboard widget. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Property	Type	Description
relative_path	string	Ścieżka do katalogu modułu względna względem katalogu frontend Zabbixa. Możliwe wartości: `widgets/*` - dla wbudowanych modułów widgetów; `modules/*` - dla modułów firm trzecich. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
status	integer	Określa, czy moduł jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* Wyłączony; 1 - Włączony.
config	object	Konfiguracja modułu.

module.create

Opis

`object module.create(object/array modules)`

Ta metoda umożliwia instalowanie nowych modułów frontend.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli parametr `$ZBX_FEATURE_FLAGS['modules_config_enabled']` ma wartość `false` w pliku konfiguracyjnym frontend, użytkownik nie będzie mógł wywołać tej metody.

Attention:
Pliki modułów należy ręcznie rozpakować w odpowiednich podkatalogach, zgodnie z właściwością `relative_path` modułów.

Parametry

`(object/array)` Moduły do utworzenia.

Metoda akceptuje moduły ze standardowymi właściwościami modułu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zainstalowanych modułów we właściwości `moduleids`. Kolejność zwracanych identyfikatorów odpowiada kolejności przekazanych modułów.

Przykłady

Instalowanie modułu

Zainstaluj moduł ze statusem „Włączony”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.create",
    "params": {
        "id": "example_module",
        "relative_path": "modules/example_module",
        "status": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

Zobacz również

- Moduł
- Moduły frontendowe

Źródło

CModule::create() w *ui/include/classes/api/services/CModule.php*.

module.delete

Opis

`object module.delete(array moduleids)`

Ta metoda umożliwia odinstalowanie modułów.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli parametr `$ZBX_FEATURE_FLAGS['modules_config_enabled']` ma wartość `false` w pliku konfiguracji frontendu, użytkownik nie będzie mógł wywołać tej metody.

Attention:
Pliki modułów należy usunąć ręcznie.

Parametry

`(array)` ID modułów do odinstalowania.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory odinstalowanych modułów we właściwości `moduleids`.

Przykłady

Odinstalowywanie wielu modułów

Odinstaluj moduły „27” i „28”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.delete",
    "params": [
        "27",
        "28"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "27",
```

```
            "28"
        ]
    },
    "id": 1
}
```

Źródło

CModule::delete() w *ui/include/classes/api/services/CModule.php*.

module.get

Opis

`integer/array module.get(object parameters)`

Metoda umożliwia pobieranie modułów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli w pliku konfiguracyjnym frontendu wartość `$ZBX_FEATURE_FLAGS['modules_config_enabled']` jest ustawiona na `false`, użytkownik nie będzie mógł wywołać tej metody.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
moduleids	ID/array	Zwraca tylko moduły o podanych identyfikatorach.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `moduleid`, `relative_path`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie modułu według ID

Pobierz wszystkie dane o modułach „1”, „2” i „25”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.get",
    "params": {
```

```
        "output": "extend",
        "moduleids": [
            "1",
            "2",
            "25"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Moduł
- Widget pulpitu
- Moduły frontend

Źródło

CModule::get() w *ui/include/classes/api/services/CModule.php*.

module.update

Opis

`object module.update(object/array modules)`

Ta metoda umożliwia aktualizowanie istniejących modułów.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.
 Jeśli parametr `$ZBX_FEATURE_FLAGS['modules_config_enabled']` ma wartość `false` w pliku konfiguracyjnym frontend, użytkownik nie będzie mógł wywołać tej metody.

Parametry

(object/array) Właściwości modułu do zaktualizowania.

Właściwość moduleid musi być zdefiniowana dla każdego modułu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko określone właściwości.

Metoda akceptuje moduły ze standardowymi właściwościami modułu.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych modułów we właściwości moduleids.

Przykłady

Wyłączanie modułu

Wyłącz moduł „25”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "module.update",
    "params": {
        "moduleid": "25",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "moduleids": [
            "25"
        ]
    },
    "id": 1
}
```

Zobacz także

- Moduł
- Moduły frontend

Źródło

CModule::update() w *ui/include/classes/api/services/CModule.php*.

Obraz

Ta klasa została zaprojektowana do pracy z obrazami.

Odwołania do obiektów:

- Image

Dostępne metody:

- image.create - tworzenie nowych obrazów
- image.delete - usuwanie obrazów
- image.get - pobieranie obrazów
- image.update - aktualizacja obrazów

Obiekt obrazu

Następujące obiekty są bezpośrednio powiązane z API image.

Obraz

Obiekt obrazu ma następujące właściwości.

Właściwość	Typ	Opis
imageid	ID	ID obrazu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa obrazu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
imagetype	integer	Typ obrazu. Możliwe wartości: 1 - *(domyślnie)* ikona; 2 - obraz tła. Zachowanie właściwości: - *stałe* - *wymagane* dla operacji tworzenia
image	string	Obraz zakodowany w Base64. Maksymalny rozmiar zakodowanego obrazu wynosi 1 MB. Maksymalny rozmiar można dostosować, zmieniając wartość stałej `ZBX_MAX_IMAGE_SIZE`. Obsługiwane formaty obrazów: PNG, JPEG, GIF i WebP. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

image.create

Opis

`object image.create(object/array images)`

Ta metoda umożliwia tworzenie nowych obrazów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Obrazy do utworzenia.

Metoda akceptuje obrazy ze standardowymi właściwościami obrazu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych obrazów we właściwości `imageids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych obrazów.

Przykłady

Utwórz obraz ikony

Utwórz ikonę chmury.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 1,
        "name": "Cloud_(24)",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAAApgB
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "188"
        ]
    },
    "id": 1
}
```

Utwórz obraz tła

Utwórz mały obraz tła, który będzie używany jako wzorzec.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "image.create",
    "params": {
        "imagetype": 2,
        "name": "background_image",
        "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAAYCAIAAABvFaqvAAABhWlDQ1BJQ0MgcHJvZmlsZQAAKJF9kb1Lw0AYxp+m
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "189"
        ]
    },
    "id": 1
}
```

Źródło

CImage::create() w *ui/include/classes/api/services/CImage.php*.

image.delete

Opis

`object image.delete(array imageIds)`

Ta metoda umożliwia usuwanie obrazów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` IDs obrazów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych obrazów w właściwości `imageids`.

Przykłady

Usuń wiele obrazów

Usuń dwa obrazy.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "image.delete",
"params": [
"188",
"192"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"imageids": [
"188",
"192"
]
},
"id": 1
}
```

Źródło

CImage::delete() w *ui/include/classes/api/services/CImage.php*.

image.get

Opis

`integer/array image.get(object parameters)`

Metoda umożliwia pobieranie obrazów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
imageids	ID/array	Zwraca tylko obrazy o podanych identyfikatorach.
sysmapids	ID/array	Zwraca obrazy używane na podanych mapach.
select_image	flag	Zwraca właściwość `image` zawierającą obraz zakodowany w Base64.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `imageid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	

Parametr	Typ	Opis
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie obrazu

Pobierz wszystkie dane obrazu o ID „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "image.get",
    "params": {
        "output": "extend",
        "select_image": true,
        "imageids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "imageid": "2",
            "imagetype": "1",
            "name": "Cloud_(24)",
            "image": "iVBORw0KGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAA
        }
    ],
    "id": 1
}
```

Źródło

CImage::get() w *ui/include/classes/api/services/CImage.php*.

image.update

Opis

`object image.update(object/array images)`

Ta metoda umożliwia aktualizację istniejących obrazów.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości obrazu do zaktualizowania.

Właściwość `imageid` musi być zdefiniowana dla każdego obrazu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje obrazy ze standardowymi właściwościami obrazu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych obrazów we właściwości `imageids`.

Przykłady

Zmień nazwę obrazu

Zmień nazwę obrazu na „Cloud icon”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "image.update",
    "params": {
        "imageid": "2",
        "name": "Cloud icon"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "imageids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CImage::update() w *ui/include/classes/api/services/CImage.php*.

Porządkowanie

Ta klasa jest zaprojektowana do pracy z porządkowaniem.

Referencje obiektu:

- Housekeeping

Dostępne metody:

- housekeeping.get - retrieve housekeeping
- housekeeping.update - update housekeeping

Obiekt sprzątania (housekeeping)

Następujące obiekty są bezpośrednio powiązane z API `housekeeping`.

Czyszczenie danych

Obiekt ustawień ma następujące właściwości.

Property	Type	Description
hk_events_mode	integer	Włącz wewnętrzne czyszczenie danych dotyczących zdarzeń i alertów. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
hk_events_trigger	string	Okres przechowywania danych wyzwalaczy. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 365d.
hk_events_service	string	Okres przechowywania danych usług. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 1d.
hk_events_internal	string	Okres przechowywania danych wewnętrznych. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 1d.
hk_events_discovery	string	Okres przechowywania danych wykrywania sieci. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 1d.
hk_events_autoreg	string	Okres przechowywania danych automatycznej rejestracji. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 1d.
hk_services_mode	integer	Włącz wewnętrzne czyszczenie danych dotyczących usług. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
hk_services	string	Okres przechowywania danych usług. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 365d.
hk_audit_mode	integer	Włącz wewnętrzne czyszczenie danych audytu. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
hk_audit	string	Okres przechowywania danych audytu. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 31d.
hk_sessions_mode	integer	Włącz wewnętrzne czyszczenie danych sesji. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
hk_sessions	string	Okres przechowywania danych sesji. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 31d.
hk_history_mode	integer	Włącz wewnętrzne czyszczenie danych historii. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.

Property	Type	Description
hk_history_global	integer	Zastąp okres przechowywania historii pozycji. Możliwe wartości: 0 - Nie zastępuj; 1 - *(domyślnie)* Zastąp.
hk_history	string	Okres przechowywania danych historii. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 31d.
hk_trends_mode	integer	Włącz wewnętrzne czyszczenie danych trendów. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
hk_trends_global	integer	Zastąp okres przechowywania trendów pozycji. Możliwe wartości: 0 - Nie zastępuj; 1 - *(domyślnie)* Zastąp.
hk_trends	string	Okres przechowywania danych trendów. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 365d.
db_extension	string	Flaga konfiguracji rozszerzenia DB. Jeśli ta flaga ma wartość ”timescaledb”, serwer zmienia swoje działanie w zakresie czyszczenia danych i usuwania pozycji. Zachowanie właściwości: - *tylko do odczytu*
compression_availability	integer	Czy baza danych (lub jej rozszerzenie) obsługuje kompresję danych. Możliwe wartości: 0 - Niedostępna; 1 - Dostępna. Zachowanie właściwości: - *tylko do odczytu*
compression_status	integer	Włącz kompresję TimescaleDB dla historii i trendów. Możliwe wartości: 0 - *(domyślnie)* Wyłączona; 1 - Włączona.
compress_older	string	Kompresuj rekordy historii i trendów starsze niż określony okres. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Domyślnie: 7d.

housekeeping.get

Opis

`object housekeeping.get(object parameters)`

Metoda umożliwia pobranie obiektu housekeeping zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Typ	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Wartości zwracane

`(object)` Zwraca obiekt housekeeping.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "housekeeping.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CHousekeeping ::get() w *ui/include/classes/api/services/CHousekeeping.php*.

housekeeping.update

Opis

`object housekeeping.update(object housekeeping)`

Ta metoda umożliwia aktualizację istniejących ustawień housekeeping.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji User roles.

Parametry

`(object)` Właściwości housekeeping do zaktualizowania.

Zwracane wartości

`(array)` Zwraca tablicę z nazwami zaktualizowanych parametrów.

Przykłady

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        "hk_events_mode",
        "hk_events_trigger",
        "hk_events_internal",
        "hk_events_discovery"
    ],
    "id": 1
}
```

Źródło

CHousekeeping::update() w *ui/include/classes/api/services/CHousekeeping.php*.

Pozycja

Ta klasa jest przeznaczona do pracy z pozycjami.

Odwołania do obiektów:

- Pozycja
  - Nagłówek HTTP
  - Pole zapytania HTTP
- Tag pozycji
- Przetwarzanie wstępne pozycji

Dostępne metody:

- item.create - utwórz nowe pozycje
- item.delete - usuń pozycje
- item.get - pobierz pozycje
- item.update - zaktualizuj pozycje

Obiekt pozycji

Następujące obiekty są bezpośrednio powiązane z API `item`.

Pozycja

Note:
Pozycji webowych nie można bezpośrednio tworzyć, aktualizować ani usuwać za pomocą Zabbix API.

Obiekt pozycji ma następujące właściwości.

Property	Type	Description
itemid	ID	Identyfikator pozycji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
delay	string	Interwał aktualizacji pozycji. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub, opcjonalnie, jeden lub więcej interwałów niestandardowych, rozdzielonych średnikami. Interwały niestandardowe mogą być kombinacją interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr ani makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisywać jako dwa makra rozdzielone ukośnikiem (np. {$FLEX_INTERVAL}/{$FLEX_PERIOD}). Przykład: 1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr Zachowanie właściwości: - *wymagane*, jeśli type ma wartość „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) lub jeśli type ma wartość „Zabbix agent (active)” (7), a key_ nie zawiera „mqtt.get”
hostid	ID	Identyfikator hosta lub szablonu, do którego należy pozycja. Zachowanie właściwości: - *stała* - *wymagane* w operacjach tworzenia
interfaceid	ID	Identyfikator interfejsu hosta pozycji. Zachowanie właściwości: - *wymagane*, jeśli pozycja należy do hosta, a type ma wartość „Zabbix agent”, „IPMI agent”, „JMX agent”, „SNMP trap” lub „SNMP agent” - *obsługiwane*, jeśli pozycja należy do hosta, a type ma wartość „Simple check”, „External check”, „SSH agent”, „TELNET agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów wykrytych
key_	string	Klucz pozycji. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych

Property	Type	Description
name	string	Nazwa pozycji. Obsługuje makra użytkownika. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
name_resolved	string	Nazwa pozycji z rozwiniętymi makrami użytkownika. Zachowanie właściwości: - *tylko do odczytu*
type	integer	Typ pozycji. Możliwe wartości: 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. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
url	string	Ciąg URL. Obsługuje makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
value_type	integer	Typ informacji pozycji. Możliwe wartości: 0 - liczba zmiennoprzecinkowa; 1 - znak; 2 - dziennik; 3 - liczba całkowita bez znaku; 4 - tekst; 5 - dane binarne; 6 - JSON. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych

Property	Type	Description
allow_traps	integer	Zezwalaj na wypełnianie wartości w sposób podobny do pozycji trappera. 0 - *(domyślnie)* Nie akceptuj danych przychodzących; 1 - Akceptuj dane przychodzące. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów wykrytych
authtype	integer	Metoda uwierzytelniania. Możliwe wartości, jeśli `type` ma wartość „SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości, jeśli `type` ma wartość „HTTP agent”: 0 - *(domyślnie)* brak; 1 - podstawowe; 2 - NTLM; 3 - Kerberos; 4 - Digest. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „SSH agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”) lub wykrytych
description	string	Opis pozycji. Zachowanie właściwości: - *tylko do odczytu* dla obiektów wykrytych
error	string	Tekst błędu, jeśli występują problemy z aktualizacją wartości pozycji. Zachowanie właściwości: - *tylko do odczytu*
flags	integer	Pochodzenie pozycji. Możliwe wartości: 0 - zwykła pozycja; 4 - pozycja przekonwertowana z prototypu. Zachowanie właściwości: - *tylko do odczytu*
follow_redirects	integer	Podążaj za przekierowaniami odpowiedzi podczas odpytywania danych. Możliwe wartości: 0 - Nie podążaj za przekierowaniami; 1 - *(domyślnie)* Podążaj za przekierowaniami. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych

Property	Type	Description
history	string	Jednostka czasu określająca, jak długo powinny być przechowywane dane historii. Akceptuje również makro użytkownika. Wartość domyślna: 31d. Zachowanie właściwości: - *tylko do odczytu* dla obiektów wykrytych
http_proxy	string	Ciąg połączenia proxy HTTP(S). Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
inventory_link	integer	Identyfikator pola inwentarza hosta wypełnianego przez pozycję. Listę obsługiwanych pól inwentarza hosta i ich identyfikatorów można znaleźć na stronie inwentarza hosta. Wartość domyślna: 0. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float”, „character”, „numeric unsigned” lub „text” - *tylko do odczytu* dla obiektów wykrytych
ipmi_sensor	string	Czujnik IPMI. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „IPMI agent”, a `key_` nie ma wartości „ipmi.get” - *obsługiwane*, jeśli `type` ma wartość „IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
jmx_endpoint	string	Niestandardowy ciąg połączenia agenta JMX. Wartość domyślna: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „JMX agent” - *tylko do odczytu* dla obiektów wykrytych
lastclock	timestamp	Czas ostatniej aktualizacji wartości pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Ten okres można wydłużyć, zmieniając wartość parametru *Max history display period* w sekcji menu *Administration > General*. Zachowanie właściwości: - *tylko do odczytu*
lastns	integer	Nanosekundy czasu ostatniej aktualizacji wartości pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Ten okres można wydłużyć, zmieniając wartość parametru *Max history display period* w sekcji menu *Administration > General*. Zachowanie właściwości: - *tylko do odczytu*

Property	Type	Description
lastvalue	string	Ostatnia wartość pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Ten okres można wydłużyć, zmieniając wartość parametru *Max history display period* w sekcji menu *Administration > General*. Zachowanie właściwości: - *tylko do odczytu*
logtimefmt	string	Format czasu we wpisach dziennika. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „log” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
master_itemid	ID	Identyfikator nadrzędnej pozycji. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
output_format	integer	Czy odpowiedź powinna zostać przekonwertowana do formatu JSON. 0 - *(domyślnie)* Przechowuj dane surowe; 1 - Konwertuj do formatu JSON. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
params	string	Dodatkowe parametry zależne od typu pozycji: - wykonywany skrypt dla pozycji SSH agent i TELNET agent; - zapytanie SQL dla pozycji database monitor; - formuła dla pozycji calculated; - skrypt dla pozycji script i browser. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Database monitor”, „SSH agent”, „TELNET agent”, „Calculated”, „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „Script” lub „Browser”) lub wykrytych
parameters	object/array	Dodatkowe parametry, jeśli `type` ma wartość „Script” lub „Browser”. Tablica obiektów z właściwościami `name` i `value`, przy czym `name` musi być unikatowe. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
password	string	Hasło do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „JMX agent”, a `username` jest ustawione - *obsługiwane*, jeśli `type` ma wartość „Simple check”, „SSH agent”, „TELNET agent”, „Database monitor” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”) lub wykrytych

Property	Type	Description
post_type	integer	Typ treści danych POST przechowywanej we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* Dane surowe; 2 - Dane JSON; 3 - Dane XML. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
posts	string	Dane treści żądania HTTP(S). Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „JSON data” lub „XML data” - *obsługiwane*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „Raw data” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
prevvalue	string	Poprzednia wartość pozycji. Domyślnie wyświetlane są tylko wartości z ostatnich 24 godzin. Ten okres można wydłużyć, zmieniając wartość parametru *Max history display period* w sekcji menu *Administration > General*. Zachowanie właściwości: - *tylko do odczytu*
privatekey	string	Nazwa pliku klucza prywatnego. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key” - *tylko do odczytu* dla obiektów wykrytych
publickey	string	Nazwa pliku klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key” - *tylko do odczytu* dla obiektów wykrytych
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych

Property	Type	Description
retrieve_mode	integer	Część odpowiedzi, która powinna zostać zapisana. Możliwe wartości, jeśli `request_method` ma wartość „GET”, „POST” lub „PUT”: 0 - *(domyślnie)* Treść; 1 - Nagłówki; 2 - Zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości, jeśli `request_method` ma wartość „HEAD”: 1 - Nagłówki. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
snmp_oid	string	OID SNMP. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
ssl_key_password	string	Hasło do pliku klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
state	integer	Stan pozycji. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - nieobsługiwana. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Stan pozycji. Możliwe wartości: 0 - *(domyślnie)* pozycja włączona; 1 - pozycja wyłączona.
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielone przecinkami. Obsługuje również makra użytkownika jako część listy rozdzielanej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych

Property	Type	Description
templateid	ID	Identyfikator pozycji nadrzędnego szablonu. *Wskazówka*: Użyj właściwości `hostid`, aby określić szablon, do którego należy pozycja. Zachowanie właściwości: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `300s`, `1m`). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-600s. Wartość domyślna: "" - użyj ustawień proxy/globalnych. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Zabbix agent" (0), „Simple check" (3), a `key_` nie zaczyna się od „`vmware.`" ani „`icmpping`", „Zabbix agent (active)" (7), „External check" (10), „Database monitor" (11), „SSH agent" (13), „TELNET agent" (14), „HTTP agent" (19), „SNMP agent" (20), a `snmp_oid` zaczyna się od „`walk[`" lub „`get[`", „Script" (21), „Browser" (22) - *tylko do odczytu* dla obiektów dziedziczonych i wykrytych, - jeśli zastosowana jest wartość domyślna, a pozycja należy do hosta monitorowanego przez niedostępny proxy lub grupę proxy, używany jest globalny limit czasu.
trapper_hosts	string	Dozwolone hosty. Zachowanie właściwości: - *tylko do odczytu* dla obiektów wykrytych - *obsługiwane*, jeśli `type` ma wartość „Zabbix trapper" lub jeśli `type` ma wartość „HTTP agent", a `allow_traps` ma wartość „Allow incoming data". Jeśli właściwość zostanie pominięta, jest automatycznie ustawiana na makro użytkownika `{$TRAPPER.ALLOWED_HOSTS}` (wartość domyślna). Domyślnie `{$TRAPPER.ALLOWED_HOSTS}` ma wartość `127.0.0.1,::1`. Jeśli właściwość zostanie ustawiona na pusty ciąg, połączenia przychodzące ze wszystkich hostów będą odrzucane.
trends	string	Jednostka czasu określająca, jak długo powinny być przechowywane dane trendów. Akceptuje również makro użytkownika. Wartość domyślna: 365d. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float" lub „numeric unsigned" - *tylko do odczytu* dla obiektów wykrytych
units	string	Jednostki wartości. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float" lub „numeric unsigned" - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych

Property	Type	Description
username	string	Nazwa użytkownika do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, „TELNET agent” lub jeśli `type` ma wartość „JMX agent”, a `password` jest ustawione - *obsługiwane*, jeśli `type` ma wartość „Simple check”, „Database monitor” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”) lub wykrytych
uuid	string	Uniwersalny unikatowy identyfikator używany do powiązania importowanej pozycji z już istniejącymi pozycjami. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwane*, jeśli pozycja należy do szablonu
valuemapid	ID	Identyfikator powiązanej mapy wartości. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float”, „character” lub „numeric unsigned” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
verify_host	integer	Czy należy sprawdzać, czy nazwa hosta używana dla połączenia jest zgodna z nazwą w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* Nie sprawdzaj; 1 - Sprawdzaj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
verify_peer	integer	Czy należy sprawdzać, czy certyfikat hosta jest autentyczny. Możliwe wartości: 0 - *(domyślnie)* Nie sprawdzaj; 1 - Sprawdzaj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
query	object	Konfiguracja zapytania telemetrycznego. Określa, jakie dane pozycja pobiera ze źródła danych telemetrycznych, jak je filtruje oraz jak agreguje wynik. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Telemetry query” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
time_shift	string	Jak daleko wstecz (przed bieżącym czasem) kończy się zakres czasu, z którego pozycja pobiera dane. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`). Akceptuje również makra użytkownika. Zakres możliwych wartości: 0-86400s. Wartość domyślna: 15s. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Telemetry query”

Property	Type	Description
lookback_limit	string	Jak daleko wstecz (przed bieżącym czasem) pozycja pobiera dane, gdy nie ma poprzedniego zapytania, od którego można kontynuować (np. przy pierwszym uruchomieniu pozycji), lub gdy poprzednie zapytanie jest starsze niż `delay` (np. z powodu niedostępności serwera Zabbix). Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-259200s. Wartość domyślna: 10m. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Telemetry query”
granularity	string	Długość fragmentu, na który dzielony jest pełny zakres czasu zapytania. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-86400s. Wartość domyślna: 15s. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Telemetry query”

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Type	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*

Właściwość	Type	Opis
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Zapytanie telemetryczne

Obiekt zapytania telemetrycznego ma następujące właściwości.

Property	Type	Description
signal_type	integer	Typ sygnału, którego dotyczy zapytanie pozycji. Możliwe wartości: 0 - Ślady; 1 - Metryki; 2 - Dzienniki. Zachowanie właściwości: - *required*
metric_point_type	integer	Typ punktu metryki, którego dotyczy zapytanie pozycji. Możliwe wartości: 0 - Suma; 1 - Wskaźnik; 2 - Histogram; 3 - Histogram wykładniczy. Zachowanie właściwości: - *required* jeśli `signal_type` ma wartość „Metrics”
columns	array	Tablica kolumn danych telemetrycznych, których dotyczy zapytanie pozycji. Może być pusta. Zachowanie właściwości: - *required*
aggregated_columns	array	Tablica zagregowanych kolumn danych telemetrycznych, których dotyczy zapytanie pozycji, oraz sposób obliczania wartości na podstawie rekordów kolumn. Zachowanie właściwości: - *required*
filter	object	Filtr zapytania telemetrycznego dla rekordów kolumn w zapytaniu. Zachowanie właściwości: - *required*

Kolumny danych telemetrycznych

Obiekt kolumn danych telemetrycznych ma następujące właściwości.

Property	Type	Description
column	string	Nazwa kolumny danych telemetrycznych, którą odpytuje pozycja. Patrz obsługiwane kolumny danych telemetrycznych. W przypadku wielu kolumn zawierających pojedyncze wartości każda wartość `column` musi być unikatowa. W przypadku wielu kolumn zawierających pary klucz-wartość każda para `column` i `attribute_key` musi być unikatowa.

Property	Type	Description
attribute_key	string	Nazwa klucza, który odpytuje pozycja w kolumnie zawierającej pary klucz-wartość. W przypadku wielu kolumn zawierających pary klucz-wartość każda para `column` i `attribute_key` musi być unikatowa. Zachowanie właściwości: - *required*, jeśli `column` jest ustawione na kolumnę zawierającą pary klucz-wartość

Kolumny zagregowanych danych telemetrycznych

Obiekt kolumn zagregowanych danych telemetrycznych ma następujące właściwości.

Property	Type	Description
column	string	Nazwa kolumny danych telemetrycznych, którą odpytuje pozycja. Więcej informacji zawiera sekcja obsługiwane kolumny danych telemetrycznych. W przypadku wielu kolumn każda wartość `column` musi być unikatowa. Ta właściwość nie jest obsługiwana, jeśli `function` ma wartość „Count”. Zachowanie właściwości: - *wymagana*, jeśli `function` ma wartość „Min”, „Max”, „Average”, „Sum” lub „Percentile”
function	integer	Funkcja, którą pozycja stosuje do rekordów kolumny. Możliwe wartości: 1 - Min; 2 - Max; 3 - Average; 4 - Count; 5 - Sum; 8 - Percentile. Zachowanie właściwości: - *wymagana*
parameters	array	Percentyl obliczany przez pozycję na podstawie rekordów kolumny. Możliwa wartość: liczba od 0 do 100 podana jako ciąg znaków. Zachowanie właściwości: - *wymagana*, jeśli `function` ma wartość „Percentile”
alias	string	Unikatowa nazwa obliczonej wartości, używana do identyfikowania jej w wartości pozycji. Każda wartość `alias` musi być unikatowa. Zachowanie właściwości: - *wymagana*

Filtr danych telemetrycznych

Obiekt filtru danych telemetrycznych ma następujące właściwości.

Property	Type	Description
conditions	array	Warunki filtru danych telemetrycznych, które musi spełniać rekord w kolumnie danych telemetrycznych. Zachowanie właściwości: - *required*
evaltype	integer	Metoda oceny warunku filtru. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *required*
eval_formula	string	Wygenerowane wyrażenie używane do oceny warunków filtru. Wyrażenie zawiera identyfikatory odwołujące się do określonych warunków filtru za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` w przypadku filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *read-only*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do określonych warunków filtru za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać identyfikatorom zdefiniowanym w warunkach filtru: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *required* jeśli `evaltype` ma wartość „Custom expression”

Warunki filtrowania danych telemetrycznych

Obiekt warunków filtrowania danych telemetrycznych ma następujące właściwości.

Property	Type	Description
column	string	Nazwa kolumny danych telemetrycznych, do której odnosi się warunek filtrowania. Patrz obsługiwane kolumny danych telemetrycznych. Zachowanie właściwości: - *wymagana*
attribute_key	string	Nazwa klucza, do którego odnosi się warunek, w kolumnie zawierającej pary klucz-wartość. Zachowanie właściwości: - *wymagana*, jeśli `column` jest ustawiona na kolumnę zawierającą pary klucz-wartość

Property	Type	Description
operator	integer	Operator warunku filtrowania. Możliwe wartości dla kolumn zawierających pojedyncze wartości: 0 - Jest równe; 1 - Nie jest równe; 2 - Zawiera; 3 - Nie zawiera. Możliwe wartości dla kolumn zawierających pary klucz-wartość: 0 - Jest równe; 1 - Nie jest równe; 12 - Istnieje. Zachowanie właściwości: - *wymagana*
value	string	Wartość, z którą należy porównać. Zachowanie właściwości: - *wymagana*, jeśli `operator` ma wartość „Jest równe”, „Nie jest równe”, „Zawiera” lub „Nie zawiera”
formulaid	string	Dowolny unikatowy identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtrowania, ale przy kolejnym żądaniu zostanie wygenerowany ponownie. Zachowanie właściwości: - *wymagana*, jeśli `evaltype` obiektu filtra zapytania telemetrycznego ma wartość „Wyrażenie niestandardowe”

Tag pozycji

Obiekt tagu pozycji ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa tagu pozycji. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu pozycji. Zachowanie właściwości: - *tylko do odczytu*.
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon; 1 - host. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

Wstępne przetwarzanie pozycji

Obiekt wstępnego przetwarzania pozycji ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ opcji wstępnego przetwarzania. Możliwe wartości: 1 - Niestandardowy mnożnik; 2 - Przycięcie z prawej; 3 - Przycięcie z lewej; 4 - Przycięcie; 5 - Wyrażenie regularne; 6 - Wartość logiczna na dziesiętną; 7 - Ósemkowa na dziesiętną; 8 - Szesnastkowa na dziesiętną; 9 - Prosta zmiana; 10 - Zmiana na sekundę; 11 - XML XPath; 12 - JSONPath; 13 - W zakresie; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdzenie błędu w JSON; 17 - Sprawdzenie błędu w XML; 18 - Sprawdzenie błędu za pomocą wyrażenia regularnego; 19 - Odrzucenie niezmienionych; 20 - Odrzucenie niezmienionych z sygnałem kontrolnym; 21 - JavaScript; 22 - Wzorzec Prometheus; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastąpienie; 26 - Sprawdzenie nieobsługiwania; 27 - XML do JSON; 28 - Wartość SNMP walk; 29 - SNMP walk do JSON; 30 - Wartość SNMP get. Zachowanie właściwości: - *wymagana*
params	string	Dodatkowe parametry używane przez opcję wstępnego przetwarzania. Wiele parametrów jest rozdzielanych znakiem nowego wiersza (\n). Jeśli `type` ma wartość „Check unsupported”, parametry mają składnię `<scope>[\n<pattern>]`, gdzie *pattern* jest wyrażeniem regularnym, a *scope* przyjmuje jedną z wartości: -1 - dopasowanie dowolnego błędu; 0 - sprawdzenie, czy komunikat błędu pasuje do *pattern*; 1 - sprawdzenie, czy komunikat błędu nie pasuje do *pattern*. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Custom multiplier” (1), „Right trim” (2), „Left trim” (3), „Trim” (4), „Regular expression” (5), „XML XPath” (11), „JSONPath” (12), „In range” (13), „Matches regular expression” (14), „Does not match regular expression” (15), „Check for error in JSON” (16), „Check for error in XML” (17), „Check for error using regular expression” (18), „Discard unchanged with heartbeat” (20), „JavaScript” (21), „Prometheus pattern” (22), „Prometheus to JSON” (23), „CSV to JSON” (24), „Replace” (25), Check unsupported (26), „SNMP walk value” (28), „SNMP walk to JSON” (29) lub „SNMP get value” (30)

Właściwość	Typ	Opis
error_handler	integer	Typ działania używanego w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości: 0 - Komunikat błędu jest ustawiany przez serwer Zabbix; 1 - Odrzucenie wartości; 2 - Ustawienie niestandardowej wartości; 3 - Ustawienie niestandardowego komunikatu błędu. Możliwe wartości, jeśli `type` ma wartość „Check unsupported”: 1 - Odrzucenie wartości; 2 - Ustawienie niestandardowej wartości; 3 - Ustawienie niestandardowego komunikatu błędu. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Custom multiplier” (1), „Regular expression” (5), „Boolean to decimal” (6), „Octal to decimal” (7), „Hexadecimal to decimal” (8), „Simple change” (9), „Change per second” (10), „XML XPath” (11), „JSONPath” (12), „In range” (13), „Matches regular expression” (14), „Does not match regular expression” (15), „Check for error in JSON” (16), „Check for error in XML” (17), „Check for error using regular expression” (18), „Prometheus pattern” (22), „Prometheus to JSON” (23), „CSV to JSON” (24), „Check unsupported” (26), „XML to JSON” (27), „SNMP walk value” (28), „SNMP walk to JSON” (29) lub „SNMP get value” (30)
error_handler_params	string	Parametry obsługi błędów. Zachowanie właściwości: - *wymagana*, jeśli `error_handler` ma wartość „Set custom value” lub „Set custom error message”

Dla każdego typu wstępnego przetwarzania obsługiwane są następujące parametry i procedury obsługi błędów.

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
1	Niestandardowy mnożnik	liczba[1], 6^			0, 1, 2, 3
2	Przycięcie z prawej	lista znaków[2]			
3	Przycięcie z lewej	lista znaków[2]			
4	Przycięcie	lista znaków[2]			
5	Wyrażenie regularne	wzorzec[3]	wynik[2]		0, 1, 2, 3
6	Wartość logiczna na dziesiętną				0, 1, 2, 3
7	Ósemkowa na dziesiętną				0, 1, 2, 3
8	Szesnastkowa na dziesiętną				0, 1, 2, 3

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
9	Prosta zmiana				0, 1, 2, 3
10	Zmiana na sekundę				0, 1, 2, 3
11	XML XPath	ścieżka[4]			0, 1, 2, 3
12	JSONPath	ścieżka[4]			0, 1, 2, 3
13	W zakresie	min^1, 6^	max^1, 6^		0, 1, 2, 3
14	Pasuje do wyrażenia regularnego	wzorzec[3]			0, 1, 2, 3
15	Nie pasuje do wyrażenia regularnego	wzorzec[3]			0, 1, 2, 3
16	Sprawdzenie błędu w JSON	ścieżka[4]			0, 1, 2, 3
17	Sprawdzenie błędu w XML	ścieżka[4]			0, 1, 2, 3
18	Sprawdzenie błędu za pomocą wyrażenia regularnego	wzorzec[3]	wynik[2]		0, 1, 2, 3
19	Odrzucenie niezmienionych				
20	Odrzucenie niezmienionych z sygnałem kontrolnym	sekundy^5, 6^			
21	JavaScript	skrypt[2]			
22	Wzorzec Prometheus	wzorzec^6, 7^	`value`, `label`, `function`	wynik^8, 9^	0, 1, 2, 3
23	Prometheus do JSON	wzorzec^6, 7^			0, 1, 2, 3

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
24	CSV do JSON	znak[2]	znak[2]	0,1	0, 1, 2, 3
25	Zastąpienie	wyszukiwany ciąg[2]	zastąpienie[2]		
26	Sprawdzanie nieobsługiwania	zakres[1]	wzorzec^3, 6^		1, 2, 3
27	XML do JSON				0, 1, 2, 3
28	Wartość SNMP walk	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS		0, 1, 2, 3
29	SNMP walk do JSON[10]	Nazwa pola[2]	Prefiks OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS	0, 1, 2, 3
30	Wartość SNMP get	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS			0, 1, 2, 3

[1] liczba zmiennoprzecinkowa (liczby całkowite są niejawnie konwertowane na wartości zmiennoprzecinkowe)
 [2] ciąg znaków
 [3] wyrażenie regularne
 [4] JSONPath lub XML XPath
 [5] dodatnia liczba całkowita (z obsługą przyrostków czasu, np. 30s, 1m, 2h, 1d)
 [6] makro użytkownika
 [7] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy składnik wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika.
 [8] wynik Prometheus zgodny ze składnią: `<label name>` (może być makrem użytkownika), jeśli jako drugi parametr wybrano `label`.
 [9] Jedna z funkcji agregujących: `sum`, `min`, `max`, `avg`, `count`, jeśli jako drugi parametr wybrano `function`.
 [10] Obsługuje wiele rekordów „Nazwa pola,Prefiks OID,Format” rozdzielonych znakiem nowego wiersza.

item.create

Opis

`object item.create(object/array items)`

Ta metoda umożliwia tworzenie nowych pozycji.

> **Note:**
> Pozycji webowych nie można tworzyć za pomocą Zabbix API.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Pozycje do utworzenia.

Oprócz standardowych właściwości pozycji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
preprocessing	array	Opcje wstępnego przetwarzania pozycji.
tags	array	Tagi pozycji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych pozycji we właściwości `itemids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych pozycji.

Przykłady

Tworzenie pozycji

Utwórz numeryczną pozycję typu Zabbix agent z 2 tagami pozycji do monitorowania wolnego miejsca na dysku na hoście o ID „30074”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "Free disk space on /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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24758"
        ]
    },
    "id": 1
}
```

Tworzenie pozycji inwentaryzacji hosta

Utwórz pozycję agenta Zabbix, aby wypełnić pole inwentaryzacji hosta „OS”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "24759"
        ]
    },
    "id": 1
}
```

Tworzenie pozycji z przetwarzaniem wstępnym

Utwórz pozycję przy użyciu niestandardowego mnożnika.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "itemids": [
            "44210"
        ]
    },
    "id": 1
}
```

Tworzenie zależnej pozycji

Utwórz zależną pozycję dla nadrzędnej pozycji o ID 24759. Dozwolone są tylko zależności w obrębie tego samego hosta, dlatego nadrzędna i zależna pozycja powinny mieć ten sam hostid.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Utwórz pozycję HTTP agent

Utwórz pozycję z metodą żądania POST z preprocessingiem odpowiedzi JSON.

Żądanie:

```
{
    "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": "HTTP agent example JSON",
        "value_type": 0,
        "output_format": 1,
        "preprocessing": [
            {
                "type": 12,
                "params": "$.random",
                "error_handler": 0,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Utwórz pozycję skryptową

Utwórz proste zbieranie danych za pomocą pozycji skryptowej.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.create",
    "params": {
        "name": "Script example",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
```

```
    },
    "id": 1
}
```

Źródło

CItem::create() w *ui/include/classes/api/services/CItem.php*.

item.delete

Opis

`object item.delete(array itemIds)`

Ta metoda umożliwia usuwanie pozycji.

Note:
Pozycji webowych nie można usuwać za pomocą Zabbix API.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID pozycji do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych pozycji we właściwości `itemids`.

Przykłady

Usuwanie wielu pozycji

Usuń dwie pozycje.
 Pozycje zależne i prototypy pozycji są usuwane automatycznie, jeśli główna pozycja zostanie usunięta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.delete",
    "params": [
        "22982",
        "22986"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22982",
            "22986"
        ]
    },
    "id": 1
}
```

Źródło

CItem::delete() w *ui/include/classes/api/services/CItem.php*.

item.get

Opis

`integer/array item.get(object parameters)`

Metoda umożliwia pobieranie pozycji zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
itemids	ID/array	Zwraca tylko pozycje o podanych ID.
groupids	ID/array	Zwraca tylko pozycje należące do hostów z podanych grup.
templateids	ID/array	Zwraca tylko pozycje należące do podanych szablonów.
hostids	ID/array	Zwraca tylko pozycje należące do podanych hostów.
proxyids	ID/array	Zwraca tylko pozycje monitorowane przez podane proxy.
interfaceids	ID/array	Zwraca tylko pozycje używające podanych interfejsów hosta.
graphids	ID/array	Zwraca tylko pozycje używane na podanych wykresach.
triggerids	ID/array	Zwraca tylko pozycje używane w podanych wyzwalaczach.
webitems	flag	Uwzględnia pozycje web w wyniku.
inherited	boolean	Jeśli ustawione na `true`, zwraca tylko pozycje dziedziczone z szablonu.
inheritedTags	boolean	Zwraca pozycje, które mają podane `tags` również w host/szablon/podłączonych szablonach. Możliwe wartości: `true` - szablon/host/podłączone szablony również muszą mieć podane tagi; `false` - *(domyślnie)* tagi z szablonu/hosta/podłączonych szablonów są ignorowane.
templated	boolean	Jeśli ustawione na `true`, zwraca tylko pozycje należące do szablonów.
monitored	boolean	Jeśli ustawione na `true`, zwraca tylko włączone pozycje należące do monitorowanych hostów.
group	string	Zwraca tylko pozycje należące do grupy o podanej nazwie.
host	string	Zwraca tylko pozycje należące do hosta o podanej nazwie.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwraca tylko pozycje z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie pozycje. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
with_triggers	boolean	Jeśli ustawione na `true`, zwraca tylko pozycje używane w wyzwalaczach.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów, do których należy pozycja.
selectInterfaces	query	Zwraca właściwość `interfaces` z tablicą interfejsów hosta używanych przez pozycję.

Parametr	Typ	Opis
selectTriggers	query	Zwraca właściwość `triggers` z wyzwalaczami, w których używana jest pozycja. Obsługuje `count`.
selectGraphs	query	Zwraca właściwość `graphs` z wykresami zawierającymi pozycję. Obsługuje `count`.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z danymi obiektu wykrywania pozycji. Obiekt wykrywania pozycji łączy wykrytą pozycję z prototypem pozycji, z którego została wykryta. Ma następujące właściwości: `parent_itemid` - `(string)` ID prototypu pozycji, z którego pozycja została utworzona; `key_` - `(string)` klucz prototypu pozycji; `status` - `(int)` status wykrywania pozycji: 0 - *(domyślnie)* pozycja jest wykryta, 1 - pozycja nie jest już wykrywana; `ts_delete` - `(timestamp)` czas, kiedy pozycja, która nie jest już wykrywana, zostanie usunięta; `ts_disable` - `(timestamp)` czas, kiedy pozycja, która nie jest już wykrywana, zostanie wyłączona; `disable_source` - `(int)` wskaźnik określający, czy pozycja została wyłączona przez regułę LLD czy ręcznie: 0 - *(domyślnie)* wyłączona automatycznie, 1 - wyłączona przez regułę LLD.
selectDiscoveryRule	query	Zwraca właściwość `discoveryRule` z regułą LLD, która utworzyła pozycję.
selectInheritedTags	query	Zwraca właściwość `inheritedTags` z tagami znajdującymi się w szablonie/hoście/podłączonych szablonach.
selectPreprocessing	query	Zwraca właściwość `preprocessing` z opcjami przetwarzania wstępnego pozycji.
selectTags	query	Zwraca tagi pozycji we właściwości `tags`.
selectValueMap	query	Zwraca właściwość `valuemap` z mapą wartości pozycji.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy pozycja.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectGraphs` - wyniki będą sortowane według `name`; `selectTriggers` - wyniki będą sortowane według `description`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `history`, `trends`, `type`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	

Parametr	Typ	Opis
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectItemDiscovery	query	Zwraca właściwość `itemDiscovery` z obiektem wykrywania pozycji. Obiekt wykrywania pozycji łączy pozycję z prototypem pozycji, z którego została utworzona. To zapytanie jest **przestarzałe**; zamiast niego użyj `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Wyszukiwanie pozycji według klucza

Pobierz wszystkie pozycje używane w wyzwalaczach dla określonego ID hosta, które zawierają słowo „system.cpu” w kluczu pozycji, i posortuj wyniki według nazwy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10084",
        "with_triggers": true,
        "search": {
            "key_": "system.cpu"
        },
        "sortfield": "name"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Wyszukiwanie pozycji zależnych według klucza

Pobierz wszystkie pozycje zależne z hosta o ID „10116”, które mają słowo „apache” w kluczu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": "extend",
        "hostids": "10116",
        "search": {
            "key_": "apache"
        },
        "filter": {
            "type": 18
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Znajdowanie pozycji HTTP agent

Znajdź pozycję HTTP agent z typem treści POST XML dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "hostids": "10255",
        "filter": {
            "type": 19,
            "post_type": 3
        }
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "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
}
```


Pobieranie pozycji z regułami przetwarzania wstępnego

Pobierz wszystkie pozycje i ich reguły przetwarzania wstępnego dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.get",
    "params": {
        "output": ["itemid", "name", "key_"],
        "selectPreprocessing": "extend",
        "hostids": "10254"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Reguła wykrywania
- Wykres
- Host
- Interfejs hosta
- Wyzwalacz

Źródło

CItem::get() w *ui/include/classes/api/services/CItem.php*.

item.update

Opis

`object item.update(object/array items)`

Ta metoda umożliwia aktualizację istniejących pozycji.

Note:
Pozycji webowych nie można aktualizować za pośrednictwem Zabbix API.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości pozycji do zaktualizowania.

Właściwość `itemid` musi być zdefiniowana dla każdej pozycji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości pozycji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
preprocessing	array	Opcje wstępnego przetwarzania pozycji, które zastąpią bieżące opcje wstępnego przetwarzania. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych lub obiektów wykrytych
tags	array	Tagi pozycji. Zachowanie parametru: - *tylko do odczytu* dla obiektów wykrytych

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych pozycji we właściwości `itemids`.

Przykłady

Włączanie pozycji

Włącz pozycję, czyli ustaw jej status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "10092",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "10092"
        ]
    },
    "id": 1
}
```

Aktualizacja zależnej pozycji

Zaktualizuj nazwę zależnej pozycji oraz ID pozycji nadrzędnej. Dozwolone są tylko zależności w obrębie tego samego hosta, dlatego pozycja nadrzędna i zależna pozycja powinny mieć ten sam `hostid`.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "name": "Dependent item updated name",
        "master_itemid": "25562",
        "itemid": "189019"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189019"
        ]
    },
    "id": 1
}
```

Aktualizacja pozycji HTTP agent

Włącz przechwytywanie wartości pozycji.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "allow_traps": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
        ]
    },
    "id": 1
}
```

Aktualizacja pozycji z przetwarzaniem wstępnym

Zaktualizuj pozycję za pomocą reguły przetwarzania wstępnego pozycji „W zakresie”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23856",
        "preprocessing": [
            {
                "type": 13,
                "params": "\n100",
                "error_handler": 1,
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23856"
```

```
        ]
    },
    "id": 1
}
```

Aktualizowanie pozycji skryptowej

Zaktualizuj pozycję skryptową innym skryptem i usuń niepotrzebne parametry, które były używane przez poprzedni skrypt.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "item.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "params": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItem::update() w *ui/include/classes/api/services/CItem.php*.

Pozycja wykresu

Ta klasa jest przeznaczona do pracy z pozycjami wykresu.

Odwołania do obiektów:

- Pozycja wykresu

Dostępne metody:

- graphitem.get - pobieranie pozycji wykresu

Obiekt elementu wykresu

Następujące obiekty są bezpośrednio powiązane z API `graphitem`.

Element wykresu

Note:
Elementy wykresu można modyfikować wyłącznie przez API `graph`.

Obiekt elementu wykresu ma następujące właściwości.

Właściwość	Typ	Opis
gitemid	ID	ID elementu wykresu. Zachowanie właściwości: *- tylko do odczytu*

Właściwość	Typ	Opis
color	string	Kolor rysowania elementu wykresu jako szesnastkowy kod koloru. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
itemid	ID	ID pozycji. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
calc_fnc	integer	Wartość pozycji, która będzie wyświetlana. Możliwe wartości: 1 - wartość minimalna; 2 - *(domyślnie)* wartość średnia; 4 - wartość maksymalna; 7 - wszystkie wartości; 9 - ostatnia wartość, używana tylko przez wykresy kołowe i rozbite.
drawtype	integer	Styl rysowania elementu wykresu. Możliwe wartości: 0 - *(domyślnie)* linia; 1 - wypełniony obszar; 2 - pogrubiona linia; 3 - kropka; 4 - linia przerywana; 5 - linia gradientowa.
graphid	ID	ID wykresu, do którego należy element wykresu.
sortorder	integer	Pozycja pozycji na wykresie. Domyślnie: zaczyna się od „0” i zwiększa o jeden dla każdego wpisu.
type	integer	Typ elementu wykresu. Możliwe wartości: 0 - *(domyślnie)* prosty; 2 - suma wykresu, używana tylko przez wykresy kołowe i rozbite.
yaxisside	integer	Strona wykresu, po której będzie rysowana skala Y elementu wykresu. Możliwe wartości: 0 - *(domyślnie)* lewa strona; 1 - prawa strona.

graphitem.get

Opis

`integer/array graphitem.get(object parameters)`

Metoda umożliwia pobranie elementów wykresu zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwraca tylko pozycje wykresu należące do podanych wykresów.
itemids	ID/array	Zwraca tylko pozycje wykresu o podanych identyfikatorach pozycji.

Parametr	Typ	Opis
type	integer	Zwraca tylko pozycje wykresu o podanym typie. Listę obsługiwanych typów pozycji wykresu można znaleźć na stronie obiektu pozycji wykresu.
selectGraphs	query	Zwraca właściwość `graphs` zawierającą tablicę wykresów, do których należy pozycja.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `gitemid`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
limit	integer	
output	query	
preservekeys	boolean	
sortorder	string/array	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie pozycji wykresu z wykresu

Pobierz wszystkie pozycje wykresu używane na wykresie wraz z dodatkowymi informacjami o pozycji i hoście.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphitem.get",
    "params": {
        "output": "extend",
        "graphids": "387"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Wykres

Źródło

CGraphItem::get() w *ui/include/classes/api/services/CGraphItem.php*.

Problem

Ta klasa jest przeznaczona do pracy z problemami.

Odwołania do obiektów:

- Problem
  - URL typu nośnika
- Tag problemu

Dostępne metody:

- problem.get - pobieranie problemów

Obiekt problemu

Następujące obiekty są bezpośrednio powiązane z API `problem`.

Problem

Note:
Problemy są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt problemu ma następujące właściwości.

Właściwość	Typ	Opis
eventid	ID	ID zdarzenia problemu.
source	integer	Typ zdarzenia problemu. Możliwe wartości: 0 - zdarzenie utworzone przez wyzwalacz; 3 - zdarzenie wewnętrzne; 4 - zdarzenie utworzone podczas aktualizacji stanu usługi.

Właściwość	Typ	Opis
object	integer	Typ obiektu powiązanego ze zdarzeniem problemu. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz”: 0 - wyzwalacz. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie wewnętrzne”: 0 - wyzwalacz; 4 - pozycja; 5 - reguła LLD. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone podczas aktualizacji stanu usługi”: 6 - usługa.
objectid	ID	ID powiązanego obiektu.
clock	timestamp	Czas utworzenia zdarzenia problemu.
ns	integer	Nanosekundy w momencie utworzenia zdarzenia problemu.
r_eventid	ID	ID zdarzenia przywrócenia.
r_clock	timestamp	Czas utworzenia zdarzenia przywrócenia.
r_ns	integer	Nanosekundy w momencie utworzenia zdarzenia przywrócenia.
cause_eventid	ID	ID zdarzenia przyczyny.
correlationid	ID	ID reguły korelacji, jeśli to zdarzenie zostało przywrócone przez globalną regułę korelacji.
userid	ID	ID użytkownika, który zamknął problem (jeśli problem został zamknięty ręcznie).
name	string	Rozwiązana nazwa problemu.
acknowledged	integer	Stan potwierdzenia problemu. Możliwe wartości: 0 - niepotwierdzony; 1 - potwierdzony.
severity	integer	Bieżący poziom ważności problemu. Możliwe wartości: 0 - niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa.
suppressed	integer	Czy problem jest stłumiony. Możliwe wartości: 0 - problem jest w stanie normalnym; 1 - problem jest stłumiony.
opdata	string	Dane operacyjne z rozwiniętymi makrami.
urls	array	Aktywne adresy URL typu nośnika.

URL typu nośnika

Obiekt URL typu nośnika ma następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa URL zdefiniowanego typu nośnika.
url	string	Wartość URL zdefiniowanego typu nośnika.

Wyniki będą zawierać wpisy tylko dla aktywnych typów nośników z włączonym wpisem menu zdarzeń. Makra użyte we właściwościach zostaną rozwinięte, ale jeśli jedna z właściwości zawiera nierozwinięte makro, obie właściwości zostaną wykluczone z wyników. Obsługiwane makra opisano w sekcji *Obsługiwane makra*.

Znacznik problemu

Obiekt znacznika problemu ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa znacznika problemu.
value	string	Wartość znacznika problemu.

problem.get

Opis

`integer/array problem.get(object parameters)`

Metoda umożliwia pobieranie problemów zgodnie z podanymi parametrami.

Ta metoda służy do pobierania nierozwiązanych problemów. Jeśli zostanie to określone, możliwe jest również dodatkowe pobieranie ostatnio rozwiązanych problemów. Okres określający, jak dawno oznacza "ostatnio", jest zdefiniowany w *Administracja* > *Ogólne*. Problemy, które zostały rozwiązane przed tym okresem, nie są przechowywane w tabeli problemów. Aby pobrać problemy rozwiązane dalej w przeszłości, użyj metody `event.get`.

Attention:
Ta metoda może zwrócić problemy usuniętej encji, jeśli nie zostały one jeszcze usunięte przez housekeeper.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
eventids	ID/array	Zwracaj tylko problemy o podanych ID.
groupids	ID/array	Zwracaj tylko problemy utworzone przez obiekty należące do podanych grup hostów.
hostids	ID/array	Zwracaj tylko problemy utworzone przez obiekty należące do podanych hostów.
objectids	ID/array	Zwracaj tylko problemy utworzone przez podane obiekty.
source	integer	Zwracaj tylko problemy o podanym typie. Listę obsługiwanych typów zdarzeń znajdziesz na stronie obiektu zdarzenia problemu. Domyślnie: 0 - problem utworzony przez wyzwalacz.
object	integer	Zwracaj tylko problemy utworzone przez obiekty podanego typu. Listę obsługiwanych typów obiektów znajdziesz na stronie obiektu zdarzenia problemu. Domyślnie: 0 - wyzwalacz.
acknowledged	boolean	Jeśli ustawione na `true`, zwracaj tylko potwierdzone problemy.
action	integer	Zwracaj tylko problemy, dla których wykonano podane akcje aktualizacji zdarzeń. W przypadku wielu akcji użyj sumy dowolnych akceptowalnych wartości bitmapy jako maski bitowej (na przykład 34 dla potwierdzenia i wyciszenia zdarzenia).
action_userids	ID/array	Zwracaj tylko problemy o podanych ID użytkowników, którzy wykonali akcje aktualizacji zdarzenia problemu.
suppressed	boolean	Jeśli ustawione na `true`, zwracaj tylko wyciszone problemy.

Parameter	Type	Description
symptom	boolean	Jeśli ustawione na `true`, zwracaj tylko zdarzenia problemów będące symptomami.
severities	integer/array	Zwracaj tylko problemy o podanych ważnościach zdarzeń. Dotyczy tylko wtedy, gdy object jest wyzwalacz.
evaltype	integer	Metoda oceny etykiet. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwracaj tylko problemy z podanymi etykietami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie problemy. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
recent	boolean	Jeśli ustawione na `true`, zwracaj ostatnie problemy, w tym te rozwiązane w ciągu `ok_period` oraz wszystkie nierozwiązane problemy. Jeśli ustawione na `false`, zwracaj tylko nierozwiązane problemy.
eventid_from	string	Zwracaj tylko problemy o ID większym lub równym podanemu ID.
eventid_till	string	Zwracaj tylko problemy o ID mniejszym lub równym podanemu ID.
time_from	timestamp	Zwracaj tylko problemy utworzone po podanym czasie lub w tym czasie.
time_till	timestamp	Zwracaj tylko problemy utworzone przed podanym czasem lub w tym czasie.
selectAcknowledges	query	Zwracaj właściwość `acknowledges` z aktualizacjami problemu. Aktualizacje problemu są sortowane w odwrotnej kolejności chronologicznej. Obiekt aktualizacji problemu ma następujące właściwości: `acknowledgeid` - `(ID)` ID aktualizacji; `userid` - `(ID)` ID użytkownika, który zaktualizował zdarzenie; `eventid` - `(ID)` ID zaktualizowanego zdarzenia; `clock` - `(timestamp)` czas aktualizacji zdarzenia; `message` - `(string)` treść wiadomości; `action` - `(integer)`typ akcji aktualizacji (zobacz `event.acknowledge`); `old_severity` - `(integer)` ważność zdarzenia przed tą akcją aktualizacji; `new_severity` - `(integer)` ważność zdarzenia po tej akcji aktualizacji; `suppress_until` - `(timestamp)` czas, do którego zdarzenie będzie wyciszone; `taskid` - `(ID)` ID zadania, jeśli bieżące zdarzenie przechodzi zmianę rangi; `maintenanceid` - `(ID)` ID konserwacji. Obsługuje `count`.
selectTags	query	Zwracaj właściwość `tags` z etykietami problemu. Format wyjściowy: `[{"tag": "<tag>", "value": "<value>"}, ...]`.
selectSuppressionData	query	Zwracaj właściwość `suppression_data` z listą aktywnych konserwacji i ręcznych wyciszeń: `maintenanceid` - `(ID)` ID konserwacji; `userid` - `(ID)` ID użytkownika, który wyciszył problem; `suppress_until` - `(integer)` czas, do którego problem jest wyciszony.

Parameter	Type	Description
filter	object	Zwracaj tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma zostać wykonane dopasowanie. Nie obsługuje właściwości typu `text` data type.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `eventid`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie zdarzeń problemów wyzwalacza

Pobierz ostatnie zdarzenia z wyzwalacza "15112."

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```


Pobieranie problemów potwierdzonych przez określonego użytkownika

Pobieranie problemów potwierdzonych przez użytkownika o ID=10

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "problem.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "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
}
```


Zobacz także

- Alert
- Pozycja
- Host
- Reguła LLD
- Wyzwalacz

Źródło

CEvent::get() w *ui/include/classes/api/services/CProblem.php*.

Prototyp hosta

Ta klasa jest przeznaczona do pracy z prototypami hostów.

Odwołania do obiektów:

- Prototyp hosta
- Powiązanie grupy
- Prototyp grupy
- Tag prototypu hosta
- Niestandardowy interfejs
    - Szczegóły niestandardowego interfejsu

Dostępne metody:

- hostprototype.create - tworzenie nowych prototypów hostów
- hostprototype.delete - usuwanie prototypów hostów
- hostprototype.get - pobieranie prototypów hostów
- hostprototype.update - aktualizacja prototypów hostów

Obiekt prototypu hosta

Następujące obiekty są bezpośrednio powiązane z API `hostprototype`.

Prototyp hosta

Obiekt prototypu hosta ma następujące właściwości.

Właściwość	Type	Opis
hostid	ID	ID prototypu hosta. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
host	string	Nazwa techniczna prototypu hosta. Zachowanie właściwości: - *wymagane* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
name	string	Widoczna nazwa prototypu hosta. Domyślnie: wartość właściwości `host`. Zachowanie właściwości: - *tylko do odczytu* dla obiektów dziedziczonych
status	integer	Status prototypu hosta. Możliwe wartości: 0 - *(domyślnie)* monitorowany host; 1 - niemonitorowany host.
flags	integer	Pochodzenie prototypu hosta. Możliwe wartości: 2 - prototyp hosta; 6 - wykryty prototyp hosta Zachowanie właściwości: - *tylko do odczytu*
inventory_mode	integer	Tryb uzupełniania inwentarza hosta. Możliwe wartości: -1 - *(domyślnie)* wyłączony; 0 - ręczny; 1 - automatyczny.
templateid	ID	ID nadrzędnego szablonu prototypu hosta. Zachowanie właściwości: - *tylko do odczytu*
discover	integer	Status wykrywania prototypu hosta. Możliwe wartości: 0 - *(domyślnie)* nowe hosty będą wykrywane; 1 - nowe hosty nie będą wykrywane, a istniejące hosty zostaną oznaczone jako utracone.
custom_interfaces	integer	Źródło niestandardowych interfejsów dla hostów utworzonych przez prototyp hosta. Możliwe wartości: 0 - *(domyślnie)* dziedziczenie interfejsów z nadrzędnego hosta; 1 - użycie niestandardowych interfejsów prototypów hostów. Zachowanie właściwości: - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Type	Opis
uuid	string	Uniwersalny unikatowy identyfikator używany do łączenia importowanych prototypów hostów z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane* jeśli prototyp hosta należy do szablonu

Łącze grupy

Obiekt łącza grupy łączy prototyp hosta z grupą hostów. Ma następujące właściwości.

Właściwość	Type	Opis
groupid	ID	ID grupy hostów. Zachowanie właściwości: - *wymagane*

Prototyp grupy

Obiekt prototypu grupy definiuje grupę, która zostanie utworzona dla wykrytego hosta. Ma następujące właściwości.

Właściwość	Type	Opis
group_prototypeid	ID	ID prototypu grupy. Zachowanie właściwości: - *tylko do odczytu*
name	string	Nazwa prototypu grupy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Tag prototypu hosta

Obiekt tagu prototypu hosta ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu prototypu hosta. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu prototypu hosta. Zachowanie właściwości: - *tylko do odczytu*.
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

Interfejs niestandardowy

Interfejsy niestandardowe są obsługiwane, jeśli `custom_interfaces` obiektu Host prototype object jest ustawione na „użyj niestandardowych interfejsów prototypów hostów”. Obiekt interfejsu niestandardowego ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ interfejsu. Możliwe wartości: 1 - Agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Zachowanie właściwości: - *wymagane*
useip	integer	Czy połączenie powinno być nawiązywane przez IP. Możliwe wartości: 0 - połącz przy użyciu nazwy DNS hosta; 1 - połącz przy użyciu adresu IP hosta. Zachowanie właściwości: - *wymagane*
ip	string	Adres IP używany przez interfejs. Może zawierać makra. Zachowanie właściwości: - *wymagane*, jeśli `useip` jest ustawione na „połącz przy użyciu adresu IP hosta”
dns	string	Nazwa DNS używana przez interfejs. Może zawierać makra. Zachowanie właściwości: - *wymagane*, jeśli `useip` jest ustawione na „połącz przy użyciu nazwy DNS hosta”
port	string	Numer portu używany przez interfejs. Może zawierać makra użytkownika i LLD. Zachowanie właściwości: - *wymagane*
main	integer	Czy interfejs jest używany jako domyślny na hoście. Tylko jeden interfejs danego typu może być ustawiony jako domyślny na hoście. Możliwe wartości: 0 - nie jest domyślny; 1 - domyślny. Zachowanie właściwości: - *wymagane*
details	object	Dodatkowy obiekt dla szczegółów interfejsu niestandardowego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „SNMP”

Szczegóły niestandardowego interfejsu

Obiekt szczegółów ma następujące właściwości.

Właściwość	Typ	Opis
version	integer	Wersja interfejsu SNMP. Możliwe wartości: 1 - SNMPv1; 2 - SNMPv2c; 3 - SNMPv3. Zachowanie właściwości: - *wymagane*
bulk	integer	Czy używać zbiorczych żądań SNMP. Możliwe wartości: 0 - nie używaj zbiorczych żądań; 1 - *(domyślnie)* - używaj zbiorczych żądań.
community	string	Społeczność SNMP. Zachowanie właściwości: - *wymagane* jeśli `version` jest ustawione na ”SNMPv1” lub ”SNMPv2c”
max_repetitions	integer	Maksymalna wartość powtórzeń dla natywnych zbiorczych żądań SNMP (GetBulkRequest-PDUs). Używane tylko dla pozycji `discovery[]` i `walk[]` w SNMPv2 i v3. Domyślnie: 10.
securityname	string	Nazwa bezpieczeństwa SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3”
securitylevel	integer	Poziom bezpieczeństwa SNMPv3. Możliwe wartości: 0 - *(domyślnie)* - noAuthNoPriv; 1 - authNoPriv; 2 - authPriv. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3”
authpassphrase	string	Hasło uwierzytelniania SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”
privpassphrase	string	Hasło prywatności SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authPriv”
authprotocol	integer	Protokół uwierzytelniania SNMPv3. Możliwe wartości: 0 - *(domyślnie)* - MD5; 1 - SHA1; 2 - SHA224; 3 - SHA256; 4 - SHA384; 5 - SHA512. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authNoPriv” lub ”authPriv”

Właściwość	Typ	Opis
privprotocol	integer	Protokół prywatności SNMPv3. Używany tylko przez interfejsy SNMPv3. Możliwe wartości: 0 - *(domyślnie)* - DES; 1 - AES128; 2 - AES192; 3 - AES256; 4 - AES192C; 5 - AES256C. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3” i `securitylevel` jest ustawione na ”authPriv”
contextname	string	Nazwa kontekstu SNMPv3. Zachowanie właściwości: - *obsługiwane* jeśli `version` jest ustawione na ”SNMPv3”

hostprototype.create

Opis

`object hostprototype.create(object/array hostPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów hostów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy hostów do utworzenia.

Oprócz standardowych właściwości prototypu hosta, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groupLinks	array	Powiązania grup, które mają zostać utworzone dla prototypu hosta. Zachowanie parametru: - *wymagany*
ruleid	ID	ID reguły LLD, do której należy prototyp hosta. Zachowanie parametru: - *wymagany*
groupPrototypes	array	Prototypy grup, które mają zostać utworzone dla prototypu hosta.
macros	object/array	Makra użytkownika, które mają zostać utworzone dla prototypu hosta.
tags	object/array	Tagi prototypu hosta.
interfaces	object/array	Niestandardowe interfejsy prototypu hosta.
templates	object/array	Szablony, które mają zostać podlinkowane do prototypu hosta. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów hostów we właściwości `hostids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów hostów.

Przykłady

Tworzenie prototypu hosta

Utwórz prototyp hosta "{#VM.NAME}" w regule LLD "23542" z prototypem grupy "{#HV.NAME}", parą tagów "datacenter": "{#DATACENTER.NAME}" oraz niestandardowym interfejsem SNMPv2 127.0.0.1:161 ze społecznością {$SNMP_COMMUNITY}. Połącz go z grupą hostów "2".

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10103"
        ]
    },
    "id": 1
}
```

Zobacz także

- Powiązanie grupy
- Prototyp grupy
- Tag prototypu hosta
- Niestandardowy interfejs

- Makro użytkownika

Źródło

CHostPrototype::create() w *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.delete

Opis

`object hostprototype.delete(array hostPrototypeIds)`

Ta metoda umożliwia usuwanie prototypów hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID prototypów hostów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów hostów we właściwości `hostids`.

Przykłady

Usuwanie wielu prototypów hosta

Usuń dwa prototypy hosta.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "hostprototype.delete",
"params": [
"10103",
"10105"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"hostids": [
"10103",
"10105"
]
},
"id": 1
}
```

Źródło

CHostPrototype::delete() w *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.get

Opis

`integer/array hostprototype.get(object parameters)`

Metoda umożliwia pobieranie prototypów hostów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
hostids	ID/array	Zwracaj tylko prototypy hostów o podanych ID.
discoveryids	ID/array	Zwracaj tylko prototypy hostów należące do podanych reguł LLD.
inherited	boolean	Jeśli ustawiono `true`, zwracaj tylko pozycje dziedziczone z szablonu.
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania prototypu hosta. Obiekt wykrywania prototypu hosta łączy wykryty prototyp hosta z prototypem hosta, z którego został wykryty. Ma następujące właściwości: `host` - `(string)` ID hosta; `parent_hostid` - `(string)` ID prototypu hosta, z którego utworzono prototyp hosta; `status` - `(int)` status wykrywania prototypu hosta: 0 - *(domyślnie)* prototyp hosta jest wykryty, 1 - prototyp hosta nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, kiedy prototyp hosta, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, kiedy prototyp hosta, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik określający, czy prototyp hosta został wyłączony przez regułę LLD czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z regułą LLD, do której należy prototyp hosta.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp hosta.
selectInterfaces	query	Zwracaj właściwość `interfaces` z niestandardowymi interfejsami prototypu hosta.
selectGroupLinks	query	Zwracaj właściwość `groupLinks` z powiązaniami grup prototypu hosta.
selectGroupPrototypes	query	Zwracaj właściwość `groupPrototypes` z prototypami grup prototypu hosta.
selectInheritedTags	query	Zwracaj właściwość `inheritedTags` z tagami znajdującymi się w podłączonych szablonach.
selectMacros	query	Zwracaj właściwość `macros` z makrami prototypu hosta.
selectParentHost	query	Zwracaj właściwość `parentHost` z hostem, do którego należy prototyp hosta.
selectTags	query	Zwracaj właściwość `tags` z tagami prototypu hosta.
selectTemplates	query	Zwracaj właściwość `templates` z szablonami podłączonymi do prototypu hosta. Obsługuje `count`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `hostid`, `host`, `name`, `status`, `discovered`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	

Parametr	Typ	Opis
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie prototypów hostów z reguły LLD

Pobierz wszystkie prototypy hostów, ich powiązania z grupami, prototypy grup oraz tagi z reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.get",
    "params": {
        "output": "extend",
        "selectInterfaces": "extend",
        "selectGroupLinks": "extend",
        "selectGroupPrototypes": "extend",
        "selectTags": "extend",
        "discoveryids": "23554"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Powiązanie grupy
- Prototyp grupy
- Makro użytkownika

Źródło

CHostPrototype::get() w *ui/include/classes/api/services/CHostPrototype.php*.

hostprototype.update

Opis

`object hostprototype.update(object/array hostPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów hostów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu hosta do zaktualizowania.

Właściwość `hostid` musi być zdefiniowana dla każdego prototypu hosta, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu hosta, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groupLinks	array	Dowiązania grup zastępujące bieżące dowiązania grup w prototypie hosta. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
groupPrototypes	array	Prototypy grup zastępujące istniejące prototypy grup w prototypie hosta. Wszystkie prototypy grup, które nie zostały wymienione w żądaniu, zostaną usunięte. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
macros	object/array	Makra użytkownika zastępujące bieżące makra użytkownika. Wszystkie makra, które nie zostały wymienione w żądaniu, zostaną usunięte.
tags	object/array	Tagi prototypu hosta zastępujące bieżące tagi. Wszystkie tagi, które nie zostały wymienione w żądaniu, zostaną usunięte. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
interfaces	object/array	Niestandardowe interfejsy prototypu hosta zastępujące bieżące interfejsy. Obiekt niestandardowego interfejsu powinien zawierać wszystkie swoje parametry. Wszystkie interfejsy, które nie zostały wymienione w żądaniu, zostaną usunięte. Zachowanie parametru: - *obsługiwane*, jeśli `custom_interfaces` obiektu Host prototype object jest ustawione na ”use host prototypes custom interfaces” - *tylko do odczytu* dla obiektów dziedziczonych
templates	object/array	Szablony zastępujące aktualnie podlinkowane szablony. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów hostów we właściwości `hostids`.

Przykłady

Wyłączanie prototypu hosta

Wyłącz prototyp hosta, czyli ustaw jego status na „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "status": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Aktualizowanie tagów prototypu hosta

Zastąp tagi prototypu hosta nowymi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hostprototype.update",
    "params": {
        "hostid": "10092",
        "tags": [
            {
                "tag": "datacenter",
                "value": "{#DATACENTER.NAME}"
            },
            {
                "tag": "instance-type",
                "value": "{#INSTANCE_TYPE}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Aktualizowanie niestandardowych interfejsów prototypu hosta

Zastąp dziedziczone interfejsy niestandardowymi interfejsami prototypu hosta.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "hostids": [
            "10092"
        ]
    },
    "id": 1
}
```

Zobacz także

- Łącze grupy
- Prototyp grupy
- Tag prototypu hosta
- Niestandardowy interfejs
- Makro użytkownika

Źródło

CHostPrototype::update() w *ui/include/classes/api/services/CHostPrototype.php*.

Prototyp pozycji

Ta klasa jest przeznaczona do pracy z prototypami pozycji.

Odwołania do obiektów:

- Prototyp pozycji
  - Nagłówek HTTP
  - Pole zapytania HTTP
- Tag prototypu pozycji
- Przetwarzanie wstępne prototypu pozycji

Dostępne metody:

- itemprototype.create - utwórz nowe prototypy pozycji
- itemprototype.delete - usuń prototypy pozycji
- itemprototype.get - pobierz prototypy pozycji
- itemprototype.update - zaktualizuj prototypy pozycji

Obiekt prototypu pozycji

Następujące obiekty są bezpośrednio powiązane z API `itemprototype`.

Prototyp pozycji

Obiekt prototypu pozycji ma następujące właściwości.

Property	Type	Description
itemid	ID	Identyfikator prototypu pozycji. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
delay	string	Interwał aktualizacji prototypu pozycji. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub, opcjonalnie, jeden lub więcej niestandardowych interwałów, rozdzielonych średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika i makra LLD. Jeśli wartość jest używana, musi być pojedynczym makrem. Wiele makr lub makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisywać jako dwa makra rozdzielone ukośnikiem (np. {$FLEX_INTERVAL}/{$FLEX_PERIOD}). Przykład: 1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr Zachowanie właściwości: - *wymagane*, jeśli type ma wartość „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) lub jeśli type ma wartość „Zabbix agent (active)” (7), a key_ nie zawiera „mqtt.get”
hostid	ID	Identyfikator hosta, do którego należy prototyp pozycji. Zachowanie właściwości: - *stała* - *wymagane* w operacjach tworzenia
interfaceid	ID	Identyfikator interfejsu hosta prototypu pozycji. Zachowanie właściwości: - *wymagane*, jeśli prototyp pozycji należy do hosta, a type ma wartość „Zabbix agent”, „IPMI agent”, „JMX agent”, „SNMP trap” lub „SNMP agent” - *obsługiwane*, jeśli prototyp pozycji należy do hosta, a type ma wartość „Simple check”, „External check”, „SSH agent”, „TELNET agent” lub „HTTP agent”
key_	string	Klucz prototypu pozycji. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
name	string	Nazwa prototypu pozycji. Obsługuje makra użytkownika. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
type	integer	Typ prototypu pozycji. Możliwe wartości: 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. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
url	string	Ciąg URL. Obsługuje makra LLD, makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
value_type	integer	Typ informacji prototypu pozycji. Możliwe wartości: 0 - liczba zmiennoprzecinkowa; 1 - znak; 2 - dziennik; 3 - liczba całkowita bez znaku; 4 - tekst; 5 - dane binarne; 6 - JSON. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
allow_traps	integer	Zezwala na wypełnianie wartości w sposób podobny do pozycji typu trapper. 0 - *(domyślnie)* nie akceptuj danych przychodzących; 1 - akceptuj dane przychodzące. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent”

Property	Type	Description
authtype	integer	Metoda uwierzytelniania. Możliwe wartości, jeśli `type` ma wartość „SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości, jeśli `type` ma wartość „HTTP agent”: 0 - *(domyślnie)* brak; 1 - podstawowe; 2 - NTLM; 3 - Kerberos; 4 - Digest. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „SSH agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
description	string	Opis prototypu pozycji.
flags	integer	Pochodzenie prototypu pozycji. Możliwe wartości: 2 - prototyp pozycji; 6 - wykryty prototyp pozycji Zachowanie właściwości: - *tylko do odczytu*
follow_redirects	integer	Podążanie za przekierowaniami odpowiedzi podczas odpytywania danych. Możliwe wartości: 0 - nie podążaj za przekierowaniami; 1 - *(domyślnie)* podążaj za przekierowaniami. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
history	string	Jednostka czasu określająca, jak długo mają być przechowywane dane historii. Akceptuje również makro użytkownika i makro LLD. Wartość domyślna: `31d`.
http_proxy	string	Ciąg połączenia proxy HTTP(S). Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ipmi_sensor	string	Czujnik IPMI. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „IPMI agent”, a `key_` nie ma wartości „ipmi.get” - *obsługiwane*, jeśli `type` ma wartość „IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
jmx_endpoint	string	Niestandardowy ciąg połączenia agenta JMX. Wartość domyślna: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „JMX agent”
logtimefmt	string	Format czasu we wpisach dziennika. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „log” - *tylko do odczytu* dla obiektów dziedziczonych
master_itemid	ID	Identyfikator nadrzędnej pozycji. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych
output_format	integer	Czy odpowiedź powinna zostać przekonwertowana do formatu JSON. Możliwe wartości: 0 - *(domyślnie)* przechowuj dane surowe; 1 - konwertuj do formatu JSON. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
params	string	Dodatkowe parametry zależne od typu prototypu pozycji: - wykonywany skrypt dla prototypów pozycji typu SSH agent i TELNET agent; - zapytanie SQL dla prototypów pozycji typu Database monitor; - formuła dla prototypów pozycji typu Calculated; - skrypt dla prototypów pozycji typu Script i Browser. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Database monitor”, „SSH agent”, „TELNET agent”, „Calculated”, „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „Script” lub „Browser”)
parameters	object/array	Dodatkowe parametry, jeśli `type` ma wartość „Script” lub „Browser”. Tablica obiektów z właściwościami `name` i `value`, przy czym `name` musi być unikatowe. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych
password	string	Hasło do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „JMX agent”, a `username` jest ustawione - *obsługiwane*, jeśli `type` ma wartość „Simple check”, „SSH agent”, „TELNET agent”, „Database monitor” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)

Property	Type	Description
post_type	integer	Typ treści danych wysyłanych przechowywanej we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* dane surowe. 2 - dane JSON. 3 - dane XML. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
posts	string	Dane treści żądania HTTP(S). Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „JSON data” lub „XML data” - *obsługiwane*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „Raw data” - *tylko do odczytu* dla obiektów dziedziczonych
privatekey	string	Nazwa pliku klucza prywatnego. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key”
publickey	string	Nazwa pliku klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key”
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
retrieve_mode	integer	Część odpowiedzi, która powinna zostać zapisana. Możliwe wartości, jeśli `request_method` ma wartość „GET”, „POST” lub „PUT”: 0 - *(domyślnie)* treść; 1 - nagłówki; 2 - zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości, jeśli `request_method` ma wartość „HEAD”: 1 - nagłówki. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
snmp_oid	string	OID SNMP. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_password	string	Hasło do pliku klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
status	integer	Stan prototypu pozycji. Możliwe wartości: 0 - *(domyślnie)* włączony prototyp pozycji; 1 - wyłączony prototyp pozycji; 3 - nieobsługiwany prototyp pozycji.
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielone przecinkami. Obsługuje również makra użytkownika lub makra LLD jako elementy listy rozdzielonej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
templateid	ID	Identyfikator prototypu pozycji nadrzędnego szablonu. Zachowanie właściwości: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m). Akceptuje również makra użytkownika i makra LLD. Zakres możliwych wartości: 1-600s. Wartość domyślna: ”” - użyj ustawień proxy/globalnych. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Zabbix agent” (0), „Simple check” (3), a `key_` nie zaczyna się od „`vmware.`” ani „`icmpping`”, „Zabbix agent (active)” (7), „External check” (10), „Database monitor” (11), „SSH agent” (13), „TELNET agent” (14), „HTTP agent” (19), „SNMP agent” (20), a `snmp_oid` zaczyna się od „`walk[`” lub „`get[`”, „Script” (21), „Browser” (22) - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
trapper_hosts	string	Dozwolone hosty. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Zabbix trapper” lub jeśli `type` ma wartość „HTTP agent”, a `allow_traps` ma wartość „Allow incoming data”. Jeśli właściwość zostanie pominięta, zostanie automatycznie ustawiona na `{$TRAPPER.ALLOWED_HOSTS}` makro użytkownika (wartość domyślna). Domyślnie `{$TRAPPER.ALLOWED_HOSTS}` ma wartość `127.0.0.1,::1`. Jeśli właściwość zostanie ustawiona na pusty ciąg, połączenia przychodzące będą odrzucane ze wszystkich hostów.
trends	string	Jednostka czasu określająca, jak długo mają być przechowywane dane trendów. Akceptuje również makro użytkownika i makro LLD. Wartość domyślna: 365d. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float” lub „numeric unsigned”
units	string	Jednostki wartości. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float” lub „numeric unsigned” - *tylko do odczytu* dla obiektów dziedziczonych
username	string	Nazwa użytkownika do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent” lub „TELNET agent”, albo jeśli `type` ma wartość „JMX agent”, a `password` jest ustawione - *obsługiwane*, jeśli `type` ma wartość „Simple check”, „Database monitor” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
uuid	string	Uniwersalny unikatowy identyfikator używany do łączenia importowanych prototypów pozycji z już istniejącymi prototypami. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwane*, jeśli prototyp pozycji należy do szablonu
valuemapid	ID	Identyfikator powiązanej mapy wartości. Zachowanie właściwości: - *obsługiwane*, jeśli `value_type` ma wartość „numeric float”, „character” lub „numeric unsigned” - *tylko do odczytu* dla obiektów dziedziczonych
verify_host	integer	Czy należy sprawdzić, czy nazwa hosta używana dla połączenia odpowiada nazwie w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* nie sprawdzaj; 1 - sprawdzaj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
verify_peer	integer	Czy należy sprawdzić, czy certyfikat hosta jest autentyczny. Możliwe wartości: 0 - *(domyślnie)* nie sprawdzaj; 1 - sprawdzaj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
discover	integer	Stan wykrywania prototypu pozycji. Możliwe wartości: 0 - *(domyślnie)* nowe pozycje będą wykrywane; 1 - nowe pozycje nie będą wykrywane, a istniejące pozycje zostaną oznaczone jako utracone.
query	object	Konfiguracja zapytania telemetrycznego. Określa, o jakie dane pyta pozycja w źródle danych telemetrycznych, jak filtruje dane i jak agreguje wynik. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Telemetry query” - *tylko do odczytu* dla obiektów dziedziczonych lub wykrytych
time_shift	string	Jak daleko wstecz, przed bieżącym czasem, kończy się zakres czasu, o który pyta pozycja. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`). Akceptuje również makra użytkownika i makra LLD. Zakres możliwych wartości: 0-86400s. Wartość domyślna: 15s. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Telemetry query”
lookback_limit	string	Jak daleko wstecz, przed bieżącym czasem, pozycja wyszukuje dane, gdy nie ma poprzedniego zapytania, od którego można kontynuować (np. podczas pierwszego uruchomienia pozycji), lub gdy poprzednie zapytanie jest starsze niż `delay` (np. z powodu niedostępności serwera Zabbix). Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`). Akceptuje również makra użytkownika i makra LLD. Zakres możliwych wartości: 1-259200s. Wartość domyślna: 10m. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Telemetry query”

Property	Type	Description
granularity	string	Długość fragmentu, na który dzielony jest pełny zakres czasu zapytania. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d). Akceptuje również makra użytkownika i makra LLD. Zakres możliwych wartości: 1-86400s. Wartość domyślna: 15s. Zachowanie właściwości: - *obsługiwane*, jeśli type ma wartość „Telemetry query”

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Zapytanie telemetryczne

Obiekt zapytania telemetrycznego ma następujące właściwości.

Property	Type	Description
signal_type	integer	Typ sygnału, którego dotyczy zapytanie pozycji. Możliwe wartości: 0 - Ślady; 1 - Metryki; 2 - Dzienniki. Zachowanie właściwości: - *required*
metric_point_type	integer	Typ punktu metryki, którego dotyczy zapytanie pozycji. Możliwe wartości: 0 - Suma; 1 - Wskaźnik; 2 - Histogram; 3 - Histogram wykładniczy. Zachowanie właściwości: - *required* jeśli `signal_type` ma wartość „Metrics”
columns	array	Tablica kolumn danych telemetrycznych, których dotyczy zapytanie pozycji. Może być pusta. Zachowanie właściwości: - *required*
aggregated_columns	array	Tablica zagregowanych kolumn danych telemetrycznych, których dotyczy zapytanie pozycji, oraz sposób obliczania wartości na podstawie rekordów kolumn. Zachowanie właściwości: - *required*
filter	object	Filtr zapytania telemetrycznego dla rekordów kolumn w zapytaniu. Zachowanie właściwości: - *required*

Kolumny danych telemetrycznych

Obiekt kolumn danych telemetrycznych ma następujące właściwości.

Property	Type	Description
column	string	Nazwa kolumny danych telemetrycznych, którą odpytuje pozycja. Patrz obsługiwane kolumny danych telemetrycznych. W przypadku wielu kolumn zawierających pojedyncze wartości każda wartość `column` musi być unikatowa. W przypadku wielu kolumn zawierających pary klucz-wartość każda para `column` i `attribute_key` musi być unikatowa.
attribute_key	string	Nazwa klucza, który odpytuje pozycja, w kolumnie zawierającej pary klucz-wartość. W przypadku wielu kolumn zawierających pary klucz-wartość każda para `column` i `attribute_key` musi być unikatowa. Zachowanie właściwości: - *required*, jeśli `column` jest ustawione na kolumnę zawierającą pary klucz-wartość

Kolumny zagregowanych danych telemetrycznych

Obiekt kolumn zagregowanych danych telemetrycznych ma następujące właściwości.

Właściwość	Typ	Opis
column	string	Nazwa kolumny danych telemetrycznych, którą odpytuje pozycja. Więcej informacji: obsługiwane kolumny danych telemetrycznych. W przypadku wielu kolumn każda wartość `column` musi być unikatowa. Ta właściwość nie jest obsługiwana, jeśli `function` ma wartość „Count”. Zachowanie właściwości: - *wymagana*, jeśli `function` ma wartość „Min”, „Max”, „Average”, „Sum” lub „Percentile”
function	integer	Funkcja stosowana przez pozycję do rekordów kolumny. Możliwe wartości: 1 - Min; 2 - Max; 3 - Average; 4 - Count; 5 - Sum; 8 - Percentile. Zachowanie właściwości: - *wymagana*
parameters	array	Percentyl obliczany przez pozycję na podstawie rekordów kolumny. Możliwa wartość: liczba od 0 do 100 podana jako ciąg znaków. Zachowanie właściwości: - *wymagana*, jeśli `function` ma wartość „Percentile”
alias	string	Unikatowa nazwa obliczonej wartości, używana do identyfikowania jej w wartości pozycji. Każda wartość `alias` musi być unikatowa. Zachowanie właściwości: - *wymagana*

Filtr danych telemetrycznych

Obiekt filtru danych telemetrycznych ma następujące właściwości.

Właściwość	Typ	Opis
conditions	array	Warunki filtru danych telemetrycznych, które musi spełniać rekord w kolumnie danych telemetrycznych. Zachowanie właściwości: - *wymagana*
evaltype	integer	Metoda oceny warunku filtru. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagana*

Właściwość	Typ	Opis
eval_formula	string	Wygenerowane wyrażenie używane do oceny warunków filtru. Wyrażenie zawiera identyfikatory odwołujące się do określonych warunków filtru za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do określonych warunków filtru za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać identyfikatorom zdefiniowanym w warunkach filtru: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagana*, jeśli `evaltype` ma wartość „Wyrażenie niestandardowe”

Warunki filtrowania danych telemetrycznych

Obiekt warunków filtrowania danych telemetrycznych ma następujące właściwości.

Property	Type	Description
column	string	Nazwa kolumny danych telemetrycznych, do której odnosi się warunek filtrowania. Więcej informacji zawiera sekcja obsługiwane kolumny danych telemetrycznych. Zachowanie właściwości: - *wymagana*
attribute_key	string	Nazwa klucza, do którego odnosi się warunek, w kolumnie zawierającej pary klucz-wartość. Zachowanie właściwości: - *wymagana*, jeśli `column` jest ustawiona na kolumnę zawierającą pary klucz-wartość
operator	integer	Operator warunku filtrowania. Możliwe wartości dla kolumn zawierających pojedynczą wartość: 0 - Równa się; 1 - Nie równa się; 2 - Zawiera; 3 - Nie zawiera. Możliwe wartości dla kolumn zawierających pary klucz-wartość: 0 - Równa się; 1 - Nie równa się; 12 - Istnieje. Zachowanie właściwości: - *wymagana*
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagana*, jeśli `operator` jest ustawiona na „Równa się”, „Nie równa się”, „Zawiera” lub „Nie zawiera”

Property	Type	Description
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z wyrażenia niestandardowego. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtrowania, ale przy kolejnym żądaniu zostanie wygenerowany ponownie. Zachowanie właściwości: - *wymagana*, jeśli `evaltype` obiektu filtra zapytania telemetrycznego jest ustawiona na „Wyrażenie niestandardowe”

Tag prototypu pozycji

Obiekt tagu prototypu pozycji ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu prototypu pozycji. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu prototypu pozycji. Zachowanie właściwości: - *tylko do odczytu*.
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon; 1 - host. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

Wstępne przetwarzanie prototypu pozycji

Obiekt wstępnego przetwarzania prototypu pozycji ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ opcji wstępnego przetwarzania. Możliwe wartości: 1 - Niestandardowy mnożnik; 2 - Usuwanie znaków z prawej strony; 3 - Usuwanie znaków z lewej strony; 4 - Usuwanie znaków z obu stron; 5 - Wyrażenie regularne; 6 - Konwersja wartości logicznej na dziesiętną; 7 - Konwersja ósemkowej na dziesiętną; 8 - Konwersja szesnastkowej na dziesiętną; 9 - Prosta zmiana; 10 - Zmiana na sekundę; 11 - XML XPath; 12 - JSONPath; 13 - W zakresie; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdzanie błędu w JSON; 17 - Sprawdzanie błędu w XML; 18 - Sprawdzanie błędu za pomocą wyrażenia regularnego; 19 - Odrzucanie niezmienionych wartości; 20 - Odrzucanie niezmienionych wartości z sygnałem heartbeat; 21 - JavaScript; 22 - Wzorzec Prometheus; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastępowanie; 26 - Sprawdzanie wartości nieobsługiwanej; 27 - XML do JSON; 28 - Wartość SNMP walk; 29 - SNMP walk do JSON; 30 - Wartość SNMP get. Zachowanie właściwości: - *required*
params	string	Dodatkowe parametry używane przez opcję wstępnego przetwarzania. Wiele parametrów jest rozdzielanych znakiem nowego wiersza (\n). Jeśli `type` ma wartość „Check unsupported”, parametry są zgodne ze składnią `<scope>[\n<pattern>]`, gdzie *pattern* jest wyrażeniem regularnym, a *scope* przyjmuje jedną z wartości: -1 - dopasowanie dowolnego błędu; 0 - sprawdzenie, czy komunikat błędu pasuje do *pattern*; 1 - sprawdzenie, czy komunikat błędu nie pasuje do *pattern*. Zachowanie właściwości: - *required*, jeśli `type` ma wartość „Custom multiplier” (1), „Right trim” (2), „Left trim” (3), „Trim” (4), „Regular expression” (5), „XML XPath” (11), „JSONPath” (12), „In range” (13), „Matches regular expression” (14), „Does not match regular expression” (15), „Check for error in JSON” (16), „Check for error in XML” (17), „Check for error using regular expression” (18), „Discard unchanged with heartbeat” (20), „JavaScript” (21), „Prometheus pattern” (22), „Prometheus to JSON” (23), „CSV to JSON” (24), „Replace” (25), „Check unsupported” (26), „SNMP walk value” (28), „SNMP walk to JSON” (29) lub „SNMP get value” (30)

Właściwość	Typ	Opis
error_handler	integer	Typ działania używanego w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości: 0 - Komunikat błędu jest ustawiany przez serwer Zabbix; 1 - Odrzucenie wartości; 2 - Ustawienie niestandardowej wartości; 3 - Ustawienie niestandardowego komunikatu błędu. Możliwe wartości, jeśli `type` ma wartość „Check unsupported”: 1 - Odrzucenie wartości; 2 - Ustawienie niestandardowej wartości; 3 - Ustawienie niestandardowego komunikatu błędu. Zachowanie właściwości: - *required*, jeśli `type` ma wartość „Custom multiplier” (1), „Regular expression” (5), „Boolean to decimal” (6), „Octal to decimal” (7), „Hexadecimal to decimal” (8), „Simple change” (9), „Change per second” (10), „XML XPath” (11), „JSONPath” (12), „In range” (13), „Matches regular expression” (14), „Does not match regular expression” (15), „Check for error in JSON” (16), „Check for error in XML” (17), „Check for error using regular expression” (18), „Prometheus pattern” (22), „Prometheus to JSON” (23), „CSV to JSON” (24), „Check unsupported” (26), „XML to JSON” (27), „SNMP walk value” (28), „SNMP walk to JSON” (29) lub „SNMP get value” (30)
error_handler_params	string	Parametry obsługi błędów. Zachowanie właściwości: - *required*, jeśli `error_handler` ma wartość „Set custom value” lub „Set custom error message”

Dla każdego typu wstępnego przetwarzania obsługiwane są następujące parametry i procedury obsługi błędów.

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
1	Niestandardowy mnożnik	number^1, 6^			0, 1, 2, 3
2	Usuwanie znaków z prawej strony	lista znaków[2]			
3	Usuwanie znaków z lewej strony	lista znaków[2]			
4	Usuwanie znaków z obu stron	lista znaków[2]			
5	Wyrażenie regularne	pattern[3]	output[2]		0, 1, 2, 3

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
6	Konwersja wartości log-icznej na dziesiętną				0, 1, 2, 3
7	Konwersja ósemkowej na dziesiętną				0, 1, 2, 3
8	Konwersja szes-nastkowej na dziesiętną				0, 1, 2, 3
9	Prosta zmi-ana				0, 1, 2, 3
10	Zmiana na sekundę				0, 1, 2, 3
11	XML XPath	path[4]			0, 1, 2, 3
12	JSONPath	path[4]			0, 1, 2, 3
13	W za-kre-sie	min^1, 6^	max^1, 6^		0, 1, 2, 3
14	Pasuje do wyraże-nia regu-larnego	pattern[3]			0, 1, 2, 3
15	Nie pa-suje do wyraże-nia regu-larnego	pattern[3]			0, 1, 2, 3
16	Sprawdzanie błędu w JSON	path[4]			0, 1, 2, 3
17	Sprawdzanie błędu w XML	path[4]			0, 1, 2, 3
18	Sprawdzanie błędu za po-mocą wyraże-nia regu-larnego	pattern[3]	output[2]		0, 1, 2, 3
19	Odrzucanie niezmienionych wartości				

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
20	Odrzucanie niezmienionych wartości z sygnałem heartbeat	seconds^5, 6^			
21	JavaScript	script[2]			
22	Wzorzec Prometheus	pattern^6, 7^	`value`, `label`, `function`	output^8, 9^	0, 1, 2, 3
23	Prometheus do JSON	pattern^6, 7^			0, 1, 2, 3
24	CSV do JSON	character[2]	character[2]	0,1	0, 1, 2, 3
25	Zastępowanie	search string[2]	replacement[2]		
26	Sprawdzanie wartości nieobsługiwanej	scope[1]	pattern^3, 6^		1, 2, 3
27	XML do JSON				0, 1, 2, 3
28	Wartość SNMP walk	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS		0, 1, 2, 3
29	SNMP walk do JSON[10]	Field name[2]	OID prefix[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS	0, 1, 2, 3
30	Wartość SNMP get	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS			0, 1, 2, 3

[1] liczba całkowita lub zmiennoprzecinkowa
 [2] ciąg znaków
 [3] wyrażenie regularne
 [4] JSONPath lub XML XPath
 [5] dodatnia liczba całkowita (z obsługą sufiksów czasu, np. 30s, 1m, 2h, 1d)
 [6] makro użytkownika, makro LLD
 [7] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy składnik wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika lub makrem LLD.
 [8] dane wyjściowe Prometheus zgodne ze składnią: `<label name>` (może być makrem użytkownika lub makrem LLD), jeśli jako drugi parametr wybrano `label`.
 [9] Jedna z funkcji agregujących: `sum`, `min`, `max`, `avg`, `count`, jeśli jako drugi parametr wybrano `function`.
 [10] Obsługuje wiele rekordów „Field name,OID prefix,Format” rozdzielanych znakiem nowego wiersza.

itemprototype.create

Opis

`object itemprototype.create(object/array itemPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów pozycji.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototyp pozycji do utworzenia.

Oprócz standardowych właściwości prototypu pozycji, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
ruleid	ID	ID reguły LLD, do której należy pozycja. Zachowanie parametru: - *wymagany*
preprocessing	array	Opcje przetwarzania wstępnego prototypu pozycji.
tags	array	Tagi prototypu pozycji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów pozycji we właściwości `itemids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów pozycji.

Przykłady

Tworzenie prototypu pozycji

Utwórz prototyp pozycji, aby monitorować wolne miejsce na dysku w wykrytym systemie plików. Wykryte pozycje powinny być numerycznymi pozycjami agenta Zabbix, aktualizowanymi co 30 sekund.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Free disk space on {#FSNAME}",
        "key_": "vfs.fs.size[{#FSNAME},free]",
        "hostid": "10197",
        "ruleid": "27665",
        "type": 0,
        "value_type": 3,
        "interfaceid": "112",
        "delay": "30s"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27666"
        ]
    },
    "id": 1
}
```

Tworzenie prototypu pozycji z przetwarzaniem wstępnym

Utwórz pozycję, używając zmiany na sekundę oraz niestandardowego mnożnika jako drugiego kroku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Incoming network traffic on {#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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Tworzenie prototypu zależnej pozycji

Utwórz prototyp zależnej pozycji dla prototypu pozycji nadrzędnej o ID 44211. Dozwolone są tylko zależności w obrębie tego samego hosta (szablonu/reguły wykrywania), dlatego pozycja nadrzędna i zależna pozycja powinny mieć takie samo hostid i ruleid.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
      "hostid": "10001",
      "ruleid": "27665",
      "name": "Dependent test item prototype",
      "key_": "dependent.prototype",
      "type": 18,
      "master_itemid": "44211",
      "value_type": 3
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44212"
        ]
    },
    "id": 1
}
```

Utwórz prototyp pozycji HTTP agent

Utwórz prototyp pozycji z adresem URL używającym makra użytkownika, pól zapytania i niestandardowych nagłówków.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "type": "19",
        "hostid": "10254",
        "ruleid": "28256",
        "interfaceid": "2",
        "name": "api item prototype example",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Utwórz prototyp pozycji skryptowej

Utwórz proste zbieranie danych przy użyciu prototypu pozycji skryptowej.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.create",
    "params": {
        "name": "Script example",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItemPrototype::create() w *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.delete

Opis

`object itemprototype.delete(array itemPrototypeIds)`

Ta metoda umożliwia usuwanie prototypów pozycji.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID prototypów pozycji do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów pozycji we właściwości `prototypeids`.

Przykłady

Usuwanie wielu prototypów pozycji

Usuń dwa prototypy pozycji.
 Zależne prototypy pozycji są usuwane automatycznie, jeśli pozycja nadrzędna lub prototyp pozycji zostanie usunięty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.delete",
    "params": [
        "27352",
        "27356"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "prototypeids": [
            "27352",
            "27356"
        ]
    },
    "id": 1
}
```

Źródło

CItemPrototype::delete() w *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.get

Opis

`integer/array itemprototype.get(object parameters)`

Metoda umożliwia pobranie prototypów pozycji zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
discoveryids	ID/array	Zwracaj tylko prototypy pozycji należące do podanych reguł LLD.
graphids	ID/array	Zwracaj tylko prototypy pozycji używane w podanych prototypach wykresów.
hostids	ID/array	Zwracaj tylko prototypy pozycji należące do podanych hostów.
inherited	boolean	Jeśli ustawiono na `true`, zwracaj tylko prototypy pozycji odziedziczone z szablonu.
itemids	ID/array	Zwracaj tylko prototypy pozycji o podanych ID.
monitored	boolean	Jeśli ustawiono na `true`, zwracaj tylko włączone prototypy pozycji należące do monitorowanych hostów.
templated	boolean	Jeśli ustawiono na `true`, zwracaj tylko prototypy pozycji należące do szablonów.
templateids	ID/array	Zwracaj tylko prototypy pozycji należące do podanych szablonów.
triggerids	ID/array	Zwracaj tylko prototypy pozycji używane w podanych prototypach wyzwalaczy.

Parametr	Typ	Opis
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania prototypu pozycji. Obiekt wykrywania prototypu pozycji łączy wykryty prototyp pozycji z prototypem pozycji, z którego został wykryty. Ma następujące właściwości: `parent_itemid` - `(string)` ID prototypu pozycji, z którego utworzono prototyp pozycji; `key_` - `(string)` klucz prototypu pozycji; `status` - `(int)` status wykrywania prototypu pozycji: 0 - *(domyślnie)* prototyp pozycji jest wykryty, 1 - prototyp pozycji nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, kiedy prototyp pozycji, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, kiedy prototyp pozycji, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik określający, czy prototyp pozycji został wyłączony przez regułę LLD czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z regułą wykrywania niskiego poziomu, do której należy prototyp pozycji.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp pozycji.
selectGraphs	query	Zwracaj właściwość `graphs` z prototypami wykresów, w których używany jest prototyp pozycji. Obsługuje `count`.
selectHosts	query	Zwracaj właściwość `hosts` z tablicą hostów, do których należy prototyp pozycji.
selectInheritedTags	query	Zwracaj właściwość `inheritedTags` z tagami znajdującymi się na szablonie/hoście/podłączonych szablonach.
selectTags	query	Zwracaj tagi prototypu pozycji we właściwości `tags`.
selectTriggers	query	Zwracaj właściwość `triggers` z prototypami wyzwalaczy, w których używany jest prototyp pozycji. Obsługuje `count`.
selectPreprocessing	query	Zwracaj właściwość `preprocessing` z opcjami przetwarzania wstępnego prototypu pozycji.
selectValueMap	query	Zwracaj właściwość `valuemap` z mapą wartości prototypu pozycji.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy prototyp pozycji.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez subselecty. Ma zastosowanie do następujących subselectów: `selectGraphs` - wyniki będą sortowane według `name`; `selectTriggers` - wyniki będą sortowane według `description`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `type`, `status`, `history`, `trends`, `discovered`.
countOutput	boolean	Te parametry są opisane w reference commentary.
editable	boolean	

Parametr	Typ	Opis
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie prototypów pozycji z reguły LLD

Pobierz wszystkie prototypy pozycji dla określonego identyfikatora reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Znajdowanie zależnej pozycji

Znajdź jedną zależną pozycję dla określonego ID pozycji.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
    "params": {
        "output": "extend",
        "filter": {
            "type": 18,
            "master_itemid": "25545"
        },
        "limit": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Znajdowanie prototypu pozycji HTTP agent

Znajdź prototyp pozycji HTTP agent z metodą żądania HEAD dla określonego ID hosta.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.get",
```

```
    "params": {
        "hostids": "10254",
        "filter": {
            "type": 19,
            "request_method": 3
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Host
- Prototyp wykresu
- Prototyp wyzwalacza

Źródło

CItemPrototype::get() w *ui/include/classes/api/services/CItemPrototype.php*.

itemprototype.update

Opis

`object itemprototype.update(object/array itemPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów pozycji.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu pozycji do zaktualizowania.

Właściwość `itemid` musi być zdefiniowana dla każdego prototypu pozycji, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu pozycji, metoda akceptuje następujące parametry.

Parametr	Type	Opis
preprocessing	array	Opcje wstępnego przetwarzania prototypu pozycji, które zastąpią bieżące opcje wstępnego przetwarzania. Zachowanie parametrów: - *tylko do odczytu* dla obiektów dziedziczonych
tags	array	Tagi prototypu pozycji.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów pozycji we właściwości `itemids`.

Przykłady

Zmiana interfejsu prototypu pozycji

Zmień interfejs hosta, który będzie używany przez wykryte pozycje.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "27428",
        "interfaceid": "132"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27428"
        ]
    },
    "id": 1
}
```

Aktualizacja prototypu zależnej pozycji

Zaktualizuj prototyp zależnej pozycji, podając nowe ID prototypu pozycji nadrzędnej. Dozwolone są tylko zależności w obrębie tego samego hosta (szablonu/reguły wykrywania), dlatego pozycja nadrzędna i zależna pozycja powinny mieć takie samo `hostid` i `ruleid`.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "master_itemid": "25570",
        "itemid": "189030"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "189030"
        ]
    },
    "id": 1
}
```

Aktualizacja prototypu pozycji HTTP agent

Zmień pola zapytania i usuń wszystkie niestandardowe nagłówki.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid":"28305",
        "query_fields": [
            {
                "name": "random",
                "value": "qwertyuiopasdfghjklzxcvbnm"
            }
```

```
        ],
        "headers": []
    }
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28305"
        ]
    },
    "id": 1
}
```

Aktualizacja opcji przetwarzania wstępnego pozycji

Zaktualizuj prototyp pozycji za pomocą reguły przetwarzania wstępnego pozycji „Mnożnik niestandardowy”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "44211",
        "preprocessing": [
            {
                "type": 1,
                "params": "4",
                "error_handler": 2,
                "error_handler_params": "5"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Aktualizowanie prototypu pozycji skryptowej

Zaktualizuj prototyp pozycji skryptowej, używając innego skryptu, i usuń zbędne parametry, które były używane przez poprzedni skrypt.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Źródło

CItemPrototype::update() w *ui/include/classes/api/services/CItemPrototype.php*.

Prototyp reguły LLD

Ta klasa jest przeznaczona do pracy z prototypami reguł wykrywania niskiego poziomu.

Odwołania do obiektów:

- Prototyp reguły LLD
  - Nagłówek HTTP
  - Pole zapytania HTTP
- Filtr prototypu reguły LLD
  - Warunek filtra prototypu reguły LLD
- Ścieżka makra prototypu LLD
- Przetwarzanie wstępne prototypu reguły LLD
- Nadpisania prototypu reguły LLD
  - Filtr nadpisania prototypu reguły LLD
    - Warunek filtra nadpisania prototypu reguły LLD
  - Operacja nadpisania prototypu reguły LLD
    - Status operacji nadpisania prototypu reguły LLD
    - Wykrywanie operacji nadpisania prototypu reguły LLD
    - Okres operacji nadpisania prototypu reguły LLD
    - Historia operacji nadpisania prototypu reguły LLD
    - Trendy operacji nadpisania prototypu reguły LLD
    - Istotność operacji nadpisania prototypu reguły LLD
    - Tag operacji nadpisania prototypu reguły LLD
    - Szablon operacji nadpisania prototypu reguły LLD
    - Inwentarz operacji nadpisania prototypu reguły LLD

Dostępne metody:

- discoveryruleprototype.create - utwórz nowe prototypy reguł LLD
- discoveryruleprototype.delete - usuń prototypy reguł LLD
- discoveryruleprototype.get - pobierz prototypy reguł LLD
- discoveryruleprototype.update - zaktualizuj prototypy reguł LLD

Obiekt prototypu reguły LLD

Następujące obiekty są bezpośrednio powiązane z API `discoveryruleprototype`.

Prototyp LLD

Obiekt prototypu reguły wykrywania niskopoziomowego ma następujące właściwości.

Property	Type	Description
itemid	ID	Identyfikator prototypu reguły LLD. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
ruleid	ID	Identyfikator nadrzędnej reguły LLD/prototypu reguły LLD. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach tworzenia
delay	string	Interwał aktualizacji prototypu reguły LLD. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`) lub opcjonalnie jeden lub więcej niestandardowych interwałów, rozdzielonych średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr lub makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisać jako dwa makra rozdzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „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) lub „Browser” (22), albo jeśli `type` ma wartość „Zabbix agent (active)” (7), a `key_` nie zawiera „mqtt.get”
hostid	ID	Identyfikator hosta, do którego należy prototyp reguły LLD. Zachowanie właściwości: - *stała* - *wymagane* w operacjach tworzenia
flags	integer	Pochodzenie prototypu reguły LLD. Możliwe wartości: 3 - prototyp reguły wykrywania niskopoziomowego; 7 - wykryty prototyp reguły wykrywania niskopoziomowego. Zachowanie właściwości: - *tylko do odczytu*
interfaceid	ID	Identyfikator interfejsu hosta prototypu reguły LLD. Zachowanie właściwości: - *wymagane*, jeśli prototyp reguły LLD należy do hosta, a `type` ma wartość „Zabbix agent”, „IPMI agent”, „JMX agent” lub „SNMP agent” - *obsługiwane*, jeśli prototyp reguły LLD należy do hosta, a `type` ma wartość „Simple check”, „External check”, „SSH agent”, „TELNET agent” lub „HTTP agent”
key_	string	Klucz prototypu reguły LLD. Wymagane jest co najmniej jedno makro LLD. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
name	string	Nazwa prototypu reguły LLD. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
type	integer	Typ prototypu reguły LLD. Możliwe wartości: 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 - Zależna pozycja; 19 - HTTP agent; 20 - SNMP agent; 21 - Script; 22 - Browser; 23 - Zagnieżdżona. Zachowanie właściwości: - *wymagane* w operacjach tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
url	string	Ciąg URL. Obsługuje makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
allow_traps	integer	Zezwala na wypełnianie wartości w sposób podobny do pozycji typu trapper. Możliwe wartości: 0 - *(domyślnie)* Nie akceptuj danych przychodzących; 1 - Akceptuj dane przychodzące. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent”
authtype	integer	Metoda uwierzytelniania. Możliwe wartości, jeśli `type` ma wartość „SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości, jeśli `type` ma wartość „HTTP agent”: 0 - *(domyślnie)* brak; 1 - podstawowe; 2 - NTLM. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „SSH agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)

Property	Type	Description
description	string	Opis prototypu reguły LLD.
error	string	Tekst błędu, jeśli występują problemy z aktualizacją wartości prototypu reguły LLD. Zachowanie właściwości: - *tylko do odczytu*
follow_redirects	integer	Śledzenie przekierowań odpowiedzi podczas odpytywania danych. Możliwe wartości: 0 - Nie śledź przekierowań; 1 - *(domyślnie)* Śledź przekierowania. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
http_proxy	string	Ciąg połączenia z proxy HTTP(S). Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ipmi_sensor	string	Czujnik IPMI. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „IPMI agent”, a `key_` nie ma wartości „ipmi.get” - *obsługiwane*, jeśli `type` ma wartość „IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych
jmx_endpoint	string	Niestandardowy ciąg połączenia agenta JMX. Wartość domyślna: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „JMX agent”
lifetime	string	Okres, po którym pozycje, które nie są już wykrywane, zostaną usunięte. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub makro użytkownika. Wartość domyślna: 7d.
lifetime_type	integer	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: 0 - *(domyślnie)* Usuń po osiągnięciu progu okresu przechowywania; 1 - Nie usuwaj; 2 - Usuń natychmiast.
enabled_lifetime	string	Okres, po którym pozycje, które nie są już wykrywane, zostaną wyłączone. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub makro użytkownika. Wartość domyślna: 0.
enabled_lifetime_type	integer	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: 0 - Wyłącz po osiągnięciu progu okresu przechowywania; 1 - Nie wyłączaj; 2 - *(domyślnie)* Wyłącz natychmiast.

Property	Type	Description
master_itemid	ID	Identyfikator nadrzędnej pozycji. Reguła wykrywania nie może być nadrzędną pozycją dla innej reguły wykrywania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych
output_format	integer	Czy odpowiedź powinna zostać przekonwertowana do formatu JSON. Możliwe wartości: 0 - *(domyślnie)* Przechowuj dane surowe; 1 - Konwertuj do formatu JSON. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
params	string	Dodatkowe parametry zależne od typu prototypu reguły LLD: - wykonywany skrypt dla reguł LLD SSH i Telnet; - zapytanie SQL dla reguł LLD monitorowania baz danych; - formuła dla obliczanych reguł LLD; - skrypt dla reguł LLD typu script i browser. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „Database monitor”, „SSH agent”, „TELNET agent”, „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „Script” lub „Browser”)
parameters	object/array	Dodatkowe parametry, jeśli `type` ma wartość „Script” lub „Browser”. Tablica obiektów z właściwościami `name` i `value`, gdzie `name` musi być unikatowe. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych
password	string	Hasło do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „JMX agent”, a `username` jest ustawione - *obsługiwane*, jeśli `type` ma wartość „Simple check”, „Database monitor”, „SSH agent”, „TELNET agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
post_type	integer	Typ treści danych wysyłanych przechowywanej we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* Dane surowe; 2 - Dane JSON; 3 - Dane XML. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
posts	string	Dane treści żądania HTTP(S). Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „JSON data” lub „XML data” - *obsługiwane*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „Raw data” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
privatekey	string	Nazwa pliku klucza prywatnego. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key”
publickey	string	Nazwa pliku klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key”
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
retrieve_mode	integer	Część odpowiedzi, która powinna zostać zapisana. Możliwe wartości, jeśli `request_method` ma wartość „GET”, „POST” lub „PUT”: 0 - *(domyślnie)* Treść; 1 - Nagłówki; 2 - Zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości, jeśli `request_method` ma wartość „HEAD”: 1 - Nagłówki. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
snmp_oid	string	OID SNMP. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_password	string	Hasło do pliku klucza SSL. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Property	Type	Description
state	integer	Stan prototypu reguły LLD. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - nieobsługiwany. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Stan prototypu reguły LLD. Możliwe wartości: 0 - *(domyślnie)* włączony prototyp reguły LLD; 1 - wyłączony prototyp reguły LLD.
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielone przecinkami. Obsługuje również makra użytkownika jako część listy rozdzielonej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
templateid	ID	Identyfikator nadrzędnego prototypu reguły LLD szablonu. Zachowanie właściwości: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-600s. Wartość domyślna: ”” - użyj ustawień proxy/globalnych. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Zabbix agent” (0), „Simple check” (3), a `key_` nie zaczyna się od „`vmware.`” ani „`icmpping`”, „Zabbix agent (active)” (7), „External check” (10), „Database monitor” (11), „SSH agent” (13), „TELNET agent” (14), „HTTP agent” (19), „SNMP agent” (20), a `snmp_oid` zaczyna się od „`walk[`” lub „`get[`”, „Script” (21), „Browser” (22) - *tylko do odczytu* dla obiektów dziedziczonych
trapper_hosts	string	Dozwolone hosty. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „Zabbix trapper” lub jeśli `type` ma wartość „HTTP agent”, a `allow_traps` ma wartość „Allow incoming data”. Jeśli właściwość zostanie pominięta, zostanie automatycznie ustawiona na makro użytkownika `{$TRAPPER.ALLOWED_HOSTS}` (wartość domyślna). Domyślnie `{$TRAPPER.ALLOWED_HOSTS}` ma wartość `127.0.0.1,::1`. Jeśli właściwość zostanie ustawiona na pusty ciąg, połączenia przychodzące będą odrzucane ze wszystkich hostów.

Property	Type	Description
username	string	Nazwa użytkownika do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` ma wartość „SSH agent”, „TELNET agent” lub jeśli `type` ma wartość „JMX agent”, a `password` jest ustawione - *obsługiwane*, jeśli `type` ma wartość „Simple check”, „Database monitor” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
uuid	string	Uniwersalny unikatowy identyfikator używany do łączenia importowanych prototypów reguł LLD z już istniejącymi prototypami. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwane*, jeśli prototyp reguły LLD należy do szablonu
verify_host	integer	Czy należy sprawdzić, czy nazwa hosta używana do połączenia jest zgodna z nazwą w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* Nie sprawdzaj; 1 - Sprawdzaj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
verify_peer	integer	Czy należy sprawdzić autentyczność certyfikatu hosta. Możliwe wartości: 0 - *(domyślnie)* Nie sprawdzaj; 1 - Sprawdzaj. Zachowanie właściwości: - *obsługiwane*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Filtr prototypu reguły LLD

Obiekt filtra prototypu reguły LLD definiuje zestaw warunków, które mogą być używane do filtrowania wykrytych obiektów. Ma następujące właściwości:

Właściwość	Type	Opis
conditions	object/array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich występowania w formule. Zachowanie właściwości: - *wymagane*
evaltype	integer	Metoda oceny warunków filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra. Wyrażenie zawiera identyfikatory odwołujące się do określonych warunków filtra przez ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do określonych warunków filtra przez ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „wyrażenie niestandardowe”

Warunek filtra prototypu reguły LLD

Obiekt warunku filtra prototypu reguły LLD definiuje oddzielne sprawdzenie wykonywane na wartości makra LLD. Ma następujące właściwości:

Właściwość	Type	Opis
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Zachowanie właściwości: - *wymagane*
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagane*, jeśli `operator` jest ustawiony na „matches regular expression” lub „does not match regular expression”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu tych warunków. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` obiektu filtra prototypu reguły LLD jest ustawiony na „custom expression”
operator	integer	Operator warunku. Możliwe wartości: 8 - *(domyślnie)* pasuje do wyrażenia regularnego; 9 - nie pasuje do wyrażenia regularnego; 12 - istnieje; 13 - nie istnieje.

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod discoveryruleprototype.get oraz discoveryruleprototype.create.

Ścieżka makra LLD

Ścieżka makra LLD ma następujące właściwości:

Właściwość	Type	Opis
lld_macro	string	Makro LLD. Zachowanie właściwości: - *wymagane*
path	string	Selektor wartości, która zostanie przypisana do odpowiadającego jej makra. Zachowanie właściwości: - *wymagane*

Wstępne przetwarzanie prototypu reguły LLD

Obiekt wstępnego przetwarzania prototypu reguły LLD ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ opcji wstępnego przetwarzania. Możliwe wartości: 5 - Wyrażenie regularne; 11 - XML XPath; 12 - JSONPath; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdzenie błędu w JSON; 17 - Sprawdzenie błędu w XML; 20 - Odrzucenie niezmienionej wartości z użyciem interwału kontrolnego; 21 - JavaScript; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastąpienie; 27 - XML do JSON; 28 - Wartość SNMP walk; 29 - SNMP walk do JSON; 30 - Pobranie wartości SNMP. Zachowanie właściwości: - *wymagana*
params	string	Dodatkowe parametry używane przez opcję wstępnego przetwarzania. Wiele parametrów jest rozdzielanych znakiem nowego wiersza (\n). Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Wyrażenie regularne” (5), „XML XPath” (11), „JSONPath” (12), „Pasuje do wyrażenia regularnego” (14), „Nie pasuje do wyrażenia regularnego” (15), „Sprawdzenie błędu w JSON” (16), „Sprawdzenie błędu w XML” (17), „Odrzucenie niezmienionej wartości z użyciem interwału kontrolnego” (20), „JavaScript” (21), „Prometheus do JSON” (23), „CSV do JSON” (24), „Zastąpienie” (25), „Wartość SNMP walk” (28), „SNMP walk do JSON” (29) lub „Pobranie wartości SNMP” (30)
error_handler	integer	Typ działania używanego w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości: 0 - Komunikat o błędzie jest ustawiany przez serwer Zabbix; 1 - Odrzucenie wartości; 2 - Ustawienie niestandardowej wartości; 3 - Ustawienie niestandardowego komunikatu o błędzie. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Wyrażenie regularne” (5), „XML XPath” (11), „JSONPath” (12), „Pasuje do wyrażenia regularnego” (14), „Nie pasuje do wyrażenia regularnego” (15), „Sprawdzenie błędu w JSON” (16), „Sprawdzenie błędu w XML” (17), „Prometheus do JSON” (23), „CSV do JSON” (24), „XML do JSON” (27), „Wartość SNMP walk” (28), „SNMP walk do JSON” (29) lub „Pobranie wartości SNMP” (30)
error_handler_params	string	Parametry obsługi błędów. Zachowanie właściwości: - *wymagana*, jeśli `error_handler` ma wartość „Ustawienie niestandardowej wartości” lub „Ustawienie niestandardowego komunikatu o błędzie”

Dla każdego typu wstępnego przetwarzania obsługiwane są następujące parametry i procedury obsługi błędów.

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
5	Wyrażenie regularne	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	Pasuje do wyrażenia regularnego	pattern[1]			0, 1, 2, 3
15	Nie pasuje do wyrażenia regularnego	pattern[1]			0, 1, 2, 3
16	Sprawdzenie błędu w JSON	path[3]			0, 1, 2, 3
17	Sprawdzenie błędu w XML	path[3]			0, 1, 2, 3
20	Odrzucenie niezmienionej wartości z użyciem interwału kontrolnego	seconds^4, 5^			
21	JavaScript	script[2]			
23	Prometheus do JSON	pattern^5, 6^			0, 1, 2, 3
24	CSV do JSON	character[2]	character[2]	0,1	0, 1, 2, 3
25	Zastąpienie	search string[2]	replacement[2]		
27	XML do JSON				0, 1, 2, 3
28	Wartość SNMP walk	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS		0, 1, 2, 3

Typ wstępnego przetwarzania	Nazwa	Parametr 1	Parametr 2	Parametr 3	Obsługiwane procedury obsługi błędów
29	SNMP walk do JSON[7]	Field name[2]	OID prefix[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS	0, 1, 2, 3
30	Pobranie wartości SNMP	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS			0, 1, 2, 3

[1] wyrażenie regularne
 [2] ciąg znaków
 [3] JSONPath lub XML XPath
 [4] dodatnia liczba całkowita (z obsługą przyrostków czasu, np. 30s, 1m, 2h, 1d)
 [5] makro użytkownika
 [6] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy element wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika.
 [7] Obsługuje wiele rekordów „Field name,OID prefix,Format" rozdzielonych znakiem nowego wiersza.

Nadpisania prototypów reguł LLD

Obiekt nadpisań prototypów reguł LLD definiuje zestaw reguł (filtrów, warunków i operacji), które są używane do nadpisywania właściwości różnych obiektów prototypów. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Unikalna nazwa nadpisania. Zachowanie właściwości: - *wymagane*
step	integer	Unikalny numer porządkowy nadpisania. Zachowanie właściwości: - *wymagane*
stop	integer	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli wystąpi dopasowanie. Możliwe wartości: 0 - *(domyślnie)* nie zatrzymuj przetwarzania nadpisań; 1 - zatrzymaj przetwarzanie nadpisań, jeśli filtr pasuje.
filter	object	Filtr nadpisania.
operations	object/array	Operacje nadpisania.

Filtr nadpisania prototypu reguły LLD

Obiekt filtra nadpisania prototypu reguły LLD definiuje zestaw warunków, które po dopasowaniu do wykrytego obiektu powodują zastosowanie nadpisania. Ma on następujące właściwości:

Property	Type	Description
conditions	object/array	Zestaw warunków filtra nadpisania używanych do dopasowywania wykrytych obiektów. Warunki zostaną posortowane zgodnie z kolejnością ich umieszczenia w formule. Property behavior: - *required*

Property	Type	Description
evaltype	integer	Metoda oceny warunku filtra nadpisania. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Property behavior: - *required*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra nadpisania. Wyrażenie zawiera identyfikatory odwołujące się do konkretnych warunków filtra nadpisania za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Property behavior: - *read-only*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów nadpisania z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do konkretnych warunków filtra nadpisania za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra nadpisania: żaden warunek nie może pozostać nieużyty ani pominięty. Property behavior: - *required* jeśli `evaltype` jest ustawione na ”custom expression”

Warunek filtra nadpisania prototypu reguły LLD

Obiekt warunku filtra nadpisania prototypu reguły LLD definiuje osobne sprawdzenie wykonywane na wartości makra LLD. Ma następujące właściwości:

Właściwość	Type	Opis
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Property behavior: - *wymagane*
value	string	Wartość do porównania. Property behavior: - *wymagane*, jeśli `operator` jest ustawiony na „matches regular expression” lub „does not match regular expression”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu tych warunków. Property behavior: - *wymagane*, jeśli `evaltype` obiektu LLD rule prototype override filter object jest ustawione na „custom expression”
operator	integer	Operator warunku. Możliwe wartości: 8 - *(domyślnie)* matches regular expression; 9 - does not match regular expression; 12 - exists; 13 - does not exist.

Operacja nadpisania prototypu reguły LLD

Operacja nadpisania prototypu reguły LLD jest połączeniem warunków i działań wykonywanych na obiekcie prototypu. Ma następujące właściwości:

Właściwość	Type	Opis
operationobject	integer	Typ wykrytego obiektu, dla którego ma zostać wykonane działanie. Możliwe wartości: 0 - Prototyp pozycji; 1 - Prototyp wyzwalacza; 2 - Prototyp wykresu; 3 - Prototyp hosta. Zachowanie właściwości: - *wymagane*
operator	integer	Warunek nadpisania operator. Możliwe wartości: 0 - *(domyślnie)* równa się; 1 - nie równa się; 2 - zawiera; 3 - nie zawiera; 8 - pasuje; 9 - nie pasuje.
value	string	Wzorzec do dopasowania do nazwy prototypu pozycji, wyzwalacza, wykresu lub hosta, zależnie od wybranego obiektu.
opstatus	object	Obiekt statusu operacji nadpisania dla obiektów prototypu pozycji, wyzwalacza i hosta.
opdiscover	object	Obiekt statusu wykrywania operacji nadpisania (wszystkie typy obiektów).
opperiod	object	Obiekt okresu operacji nadpisania (interwału aktualizacji) dla obiektu prototypu pozycji.
ophistory	object	Obiekt historii operacji nadpisania dla obiektu prototypu pozycji.
optrends	object	Obiekt trendów operacji nadpisania dla obiektu prototypu pozycji.
opseverity	object	Obiekt ważności operacji nadpisania dla obiektu prototypu wyzwalacza.
optag	object/array	Obiekt tagu operacji nadpisania dla obiektów prototypu wyzwalacza i hosta.
optemplate	object/array	Obiekt szablonu operacji nadpisania dla obiektu prototypu hosta.
opinventory	object	Obiekt inwentaryzacji operacji nadpisania dla obiektu prototypu hosta.

Status operacji nadpisania prototypu reguły LLD

Status operacji nadpisania prototypu reguły LLD ustawiany dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Typ	Opis
status	integer	Nadpisuje status dla wybranego obiektu. Możliwe wartości: 0 - Utwórz jako włączone; 1 - Utwórz jako wyłączone. Zachowanie właściwości: - *wymagane*

Operacja discover nadpisania prototypu reguły LLD

Status discover operacji nadpisania prototypu reguły LLD jest ustawiany dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Typ	Opis
discover	integer	Nadpisuje status discover dla wybranego obiektu. Możliwe wartości: 0 - Tak, kontynuuj wykrywanie obiektów; 1 - Nie, nowe obiekty nie będą wykrywane, a istniejące zostaną oznaczone jako utracone. Zachowanie właściwości: - *wymagane*

Okres operacji nadpisania prototypu reguły LLD

Okres operacji nadpisania prototypu reguły LLD to wartość interwału aktualizacji ustawiana dla wykrytej pozycji. Ma ona następujące właściwości:

Property	Type	Description
delay	string	Nadpisuje interwał aktualizacji prototypu pozycji. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub, opcjonalnie, jeden lub więcej niestandardowych interwałów, rozdzielonych średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika lub makra LLD. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr ani makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisać jako dwa makra rozdzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *required*

Historia operacji nadpisania prototypu reguły LLD

Wartość historii operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
history	string	Nadpisuje historię prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane historii powinny być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *wymagane*

Trendy operacji nadpisania prototypu reguły LLD

Wartość trendów operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
trends	string	Nadpisuje trendy prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane trendów powinny być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *wymagane*

Ważność operacji nadpisania prototypu reguły LLD

Wartość ważności operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytego wyzwalacza. Ma następujące właściwości:

Właściwość	Type	Opis
severity	integer	Nadpisuje ważność prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa. Zachowanie właściwości: - *wymagane*

Znacznik operacji nadpisania prototypu reguły LLD

Obiekt znacznika operacji nadpisania prototypu reguły LLD zawiera nazwę i wartość znacznika, które są ustawiane dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Type	Opis
tag	string	Nowa nazwa znacznika. Zachowanie właściwości: - *wymagane*
value	string	Nowa wartość znacznika.

Szablon operacji nadpisania prototypu reguły LLD

Obiekt szablonu operacji nadpisania prototypu reguły LLD, który jest połączony z wykrytym hostem. Ma następujące właściwości:

Właściwość	Type	Opis
templateid	ID	Nadpisuje szablon szablonów połączonych z prototypem hosta. Zachowanie właściwości: - *wymagane*

Wartość trybu inwentaryzacji operacji nadpisania prototypu reguły LLD

Wartość trybu inwentaryzacji operacji nadpisania prototypu reguły LLD, która jest ustawiana dla wykrytego hosta. Ma następujące właściwości:

Właściwość	Type	Opis
inventory_mode	integer	Nadpisuje tryb inwentaryzacji prototypu hosta. Możliwe wartości: -1 - wyłączone; 0 - *(domyślnie)* ręczne; 1 - automatyczne. Zachowanie właściwości: - *wymagane*

discoveryruleprototype.create

Opis

`object discoveryruleprototype.create(object/array lldRules)`

Ta metoda umożliwia tworzenie nowych prototypów reguł LLD.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy reguł LLD do utworzenia.

Oprócz standardowych właściwości prototypu reguły LLD, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Filtr prototypu reguły LLD dla reguły LLD.
preprocessing	object/array	Opcje przetwarzania wstępnego prototypu reguły LLD.
lld_macro_paths	object/array	Opcje lld_macro_path prototypu reguły LLD.
overrides	object/array	Opcje nadpisań prototypu reguły LLD.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów reguł LLD we właściwości `itemids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów reguł LLD.

Przykłady

Tworzenie prototypu reguły LLD

Utwórz prototyp reguły LLD (Typ: Zagnieżdżony), aby wykrywać tablespace'y w instancji bazy danych.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.create",
    "params": {
        "name": "Discover tablespaces for {#DB}",
        "key_": "db.tablespace.discovery[{#DB}]",
        "hostid": "10084",
        "ruleid": "47251",
        "type": 23
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47252"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr prototypu reguły LLD
- Ścieżki makr LLD
- Wstępne przetwarzanie prototypu reguły LLD

Źródło

CDiscoveryRulePrototype::create() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.delete

Opis

`object discoveryruleprototype.delete(array lldRuleIds)`

Ta metoda umożliwia usuwanie prototypów reguł LLD.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID prototypów reguł LLD do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów reguł LLD we właściwości `ruleids`.

Przykłady

Usuwanie wielu prototypów reguł LLD

Usuń dwa prototypy reguł LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.delete",
    "params": [
        "47252",
        "47253"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "ruleids": [
            "47252",
            "47253"
        ]
    },
```

```
    "id": 1
}
```

Źródło

CDiscoveryRulePrototype::delete() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.get

Opis

`integer/array discoveryruleprototype.get(object parameters)`

Ta metoda umożliwia pobranie prototypów reguł LLD zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
discoveryids	ID/array	Zwracaj tylko prototypy reguł LLD należące do podanych reguł LLD lub prototypów reguł LLD.
itemids	ID/array	Zwracaj tylko prototypy reguł LLD o podanych identyfikatorach.
groupids	ID/array	Zwracaj tylko prototypy reguł LLD należące do hostów z podanych grup.
hostids	ID/array	Zwracaj tylko prototypy reguł LLD należące do podanych hostów.
inherited	boolean	Jeśli ustawiono `true`, zwracaj tylko prototypy reguł LLD odziedziczone z szablonu.
interfaceids	ID/array	Zwracaj tylko prototypy reguł LLD używające podanych interfejsów hosta.
monitored	boolean	Jeśli ustawiono `true`, zwracaj tylko włączone prototypy reguł LLD należące do monitorowanych hostów.
templated	boolean	Jeśli ustawiono `true`, zwracaj tylko prototypy reguł LLD należące do szablonów.
templateids	ID/array	Zwracaj tylko prototypy reguł LLD należące do podanych szablonów.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z nadrzędną regułą LLD prototypu reguły LLD. Zwróci pustą odpowiedź, jeśli element nadrzędny jest prototypem reguły wykrywania.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD dla prototypu reguły LLD. Zwróci pustą odpowiedź, jeśli element nadrzędny jest regułą wykrywania (nie prototypem).
selectDiscoveryRulePrototypes	query	Zwracaj właściwość `discoveryRulePrototypes` z podrzędnymi prototypami reguł LLD należącymi do prototypu reguły LLD. Obsługuje `count`.
selectFilter	query	Zwracaj właściwość `filter` z danymi filtra używanego przez prototyp reguły LLD.
selectGraphs	query	Zwraca właściwość `graphs` z prototypami wykresów należącymi do prototypu reguły LLD. Obsługuje `count`.
selectHostPrototypes	query	Zwracaj właściwość `hostPrototypes` z prototypami hostów należącymi do prototypu reguły LLD. Obsługuje `count`.

Parametr	Typ	Opis
selectHosts	query	Zwracaj właściwość `hosts` z tablicą hostów, do których należy prototyp reguły LLD.
selectItems	query	Zwracaj właściwość `items` z prototypami pozycji należącymi do prototypu reguły LLD. Obsługuje `count`.
selectTriggers	query	Zwracaj właściwość `triggers` z prototypami wyzwalaczy należącymi do prototypu reguły LLD. Obsługuje `count`.
selectLLDMacroPaths	query	Zwracaj właściwość `lld_macro_paths` z listą makr LLD i ścieżek do wartości przypisanych do każdego odpowiedniego makra.
selectPreprocessing	query	Zwracaj właściwość `preprocessing` z opcjami przetwarzania wstępnego prototypu reguły LLD.
selectOverrides	query	Zwracaj właściwość `lld_rule_overrides` z listą filtrów nadpisywania, warunków i operacji wykonywanych na obiektach prototypów.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - nazwa techniczna hosta, do którego należy prototyp reguły LLD.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Ma zastosowanie do następujących podzapytań: `selectItems`, `selectGraphs`, `selectTriggers`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `type`, `status`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie prototypów reguł wykrywania z hosta

Pobierz wszystkie prototypy reguł wykrywania dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
```

```
    "method": "discoveryruleprototype.get",
    "params": {
        "hostids": "10084"
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "47253",
            "type": "23",
            "snmp_oid": "",
            "hostid": "10084",
            "name": "Odkryj przestrzenie tabel dla {#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
}
```


Zobacz także

- Prototyp wykresu
- Host
- Prototyp pozycji
- Filtr prototypu reguły LLD
- Prototyp wyzwalacza

Źródło

CDiscoveryRulePrototype::get() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

discoveryruleprototype.update

Opis

`object discoveryruleprototype.update(object/array lldRules)`

Ta metoda umożliwia aktualizację istniejących prototypów reguł LLD.

Należy pamiętać, że aktualizacja już wykrytych prototypów jest ograniczona.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu reguły LLD do zaktualizowania.

Właściwość `itemid` musi być zdefiniowana dla każdego prototypu reguły LLD, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu reguły LLD, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Filtr prototypu reguły LLD zastępujący istniejący filtr.
preprocessing	object/array	Opcje preprocessingu prototypu reguły LLD zastępujące istniejące opcje preprocessingu. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
lld_macro_paths	object/array	Opcje lld_macro_path prototypu reguły LLD zastępujące istniejące opcje lld_macro_path. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
overrides	object/array	Opcje nadpisań prototypu reguły LLD zastępujące istniejące opcje nadpisań. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów reguł LLD we właściwości `itemids`.

Przykłady

Aktualizacja opcji preprocessing prototypu reguły LLD

Zaktualizuj prototyp reguły LLD za pomocą reguły preprocessing JSONPath. Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryruleprototype.update",
    "params": {
        "itemid": "47253",
        "preprocessing": [
            {
                "type": 12,
                "params": "$.tablespaces",
                "error_handler": 1
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "47253"
        ]
    },
    "id": 1
}
```

Źródło

CDiscoveryRulePrototype::update() w *ui/include/classes/api/services/CDiscoveryRulePrototype.php*.

Prototyp wykresu

Ta klasa jest przeznaczona do pracy z prototypami wykresów.

Odwołania do obiektów:

- Prototyp wykresu

Dostępne metody:

- graphprototype.create - utwórz nowe prototypy wykresów
- graphprototype.delete - usuń prototypy wykresów
- graphprototype.get - pobierz prototypy wykresów
- graphprototype.update - zaktualizuj prototypy wykresów

Obiekt prototypu grafu

Następujące obiekty są bezpośrednio powiązane z API `graphprototype`.

Prototyp wykresu

Obiekt prototypu wykresu ma następujące właściwości.

Właściwość	Typ	Opis
graphid	ID	ID prototypu wykresu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Właściwość	Typ	Opis
height	integer	Wysokość prototypu wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa prototypu wykresu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
width	integer	Szerokość prototypu wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
flags	integer	Pochodzenie prototypu wykresu. Możliwe wartości: 2 - prototyp wykresu; 6 - wykryty prototyp wykresu Zachowanie właściwości: - *tylko do odczytu*
graphtype	integer	Typ układu prototypu wykresu. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - skumulowany; 2 - kołowy; 3 - rozdzielony.
percent_left	float	Lewy percentyl. Domyślnie: 0.
percent_right	float	Prawy percentyl. Domyślnie: 0.
show_3d	integer	Czy wykryte wykresy kołowe i rozdzielone mają być wyświetlane w 3D. Możliwe wartości: 0 - *(domyślnie)* wyświetlaj w 2D; 1 - wyświetlaj w 3D.
show_legend	integer	Czy legenda ma być wyświetlana na wykrytym wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
show_work_period	integer	Czy czas pracy ma być wyświetlany na wykrytym wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
templateid	ID	ID prototypu wykresu nadrzędnego szablonu. Zachowanie właściwości: - *tylko do odczytu*
yaxismax	float	Stała maksymalna wartość osi Y.
yaxismin	float	Stała minimalna wartość osi Y.
ymax_itemid	ID	ID pozycji używanej jako maksymalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymax_type` było ustawione na ”calculated”.

Właściwość	Typ	Opis
ymax_type	integer	Metoda obliczania wartości maksymalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
ymin_itemid	ID	ID pozycji używanej jako minimalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymin_type` było ustawione na "calculated".
ymin_type	integer	Metoda obliczania wartości minimalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
discover	integer	Status wykrywania prototypu wykresu. Możliwe wartości: 0 - *(domyślnie)* nowe wykresy będą wykrywane; 1 - nowe wykresy nie będą wykrywane, a istniejące wykresy zostaną oznaczone jako utracone.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych prototypów wykresów z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli prototyp wykresu należy do szablonu

graphprototype.create

Opis

`object graphprototype.create(object/array graphPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów wykresów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy wykresów do utworzenia.

Oprócz standardowych właściwości prototypu wykresu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
gitems	array	Elementy wykresu, które mają zostać utworzone dla prototypów wykresów. Elementy wykresu mogą odwoływać się zarówno do pozycji, jak i prototypów pozycji, ale musi być obecny co najmniej jeden prototyp pozycji. Zachowanie parametrów: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów wykresów we właściwości `graphids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów wykresów.

Przykłady

Tworzenie prototypu wykresu

Utwórz prototyp wykresu z dwiema pozycjami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.create",
    "params": {
        "name": "Disk space usage {#FSNAME}",
        "width": 900,
        "height": 200,
        "gitems": [
            {
                "itemid": "22828",
                "color": "00AA00"
            },
            {
                "itemid": "22829",
                "color": "3333FF"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pozycja wykresu

Źródło

CGraphPrototype::create() w *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.delete

Opis

`object graphprototype.delete(array graphPrototypeIds)`

Ta metoda umożliwia usuwanie prototypów wykresów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametery

`(array)` ID prototypów wykresów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów wykresów we właściwości `graphids`.

Przykłady

Usuwanie wielu prototypów wykresów

Usuń dwa prototypy wykresów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "graphprototype.delete",
"params": [
"652",
"653"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"graphids": [
"652",
"653"
]
},
"id": 1
}
```

Źródło

CGraphPrototype::delete() w *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.get

Opis

`integer/array graphprototype.get(object parameters)`

Ta metoda umożliwia pobieranie prototypów wykresów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
discoveryids	ID/array	Zwracaj tylko prototypy wykresów należące do podanych reguł wykrywania.
graphids	ID/array	Zwracaj tylko prototypy wykresów o podanych ID.
groupids	ID/array	Zwracaj tylko prototypy wykresów należące do hostów lub szablonów w podanych grupach hostów lub grupach szablonów.
hostids	ID/array	Zwracaj tylko prototypy wykresów należące do podanych hostów.
inherited	boolean	Jeśli ustawione na `true`, zwracaj tylko prototypy wykresów odziedziczone z szablonu.
itemids	ID/array	Zwracaj tylko prototypy wykresów zawierające podane prototypy pozycji.
templated	boolean	Jeśli ustawione na `true`, zwracaj tylko prototypy wykresów należące do szablonów.
templateids	ID/array	Zwracaj tylko prototypy wykresów należące do podanych szablonów.

Parametr	Type	Opis
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania prototypu wykresu. Obiekt wykrywania prototypu wykresu łączy wykryty prototyp wykresu z prototypem wykresu, z którego został wykryty. Ma następujące właściwości: `parent_graphid` - `(ID)` ID prototypu wykresu, z którego wykres został utworzony; `status` - `(int)` status wykrywania wykresu: 0 - *(domyślnie)* prototyp wykresu jest wykryty, 1 - prototyp wykresu nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, kiedy prototyp wykresu, który nie jest już wykrywany, zostanie usunięty.
selectDiscoveryRule	query	Zwracaj właściwość `discoveryRule` z regułą LLD, do której należy prototyp wykresu.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp wykresu.
selectGraphItems	query	Zwracaj właściwość `gitems` z elementami wykresu używanymi w prototypie wykresu.
selectHostGroups	query	Zwracaj właściwość `hostgroups` z grupami hostów, do których należy prototyp wykresu.
selectHosts	query	Zwracaj właściwość `hosts` z hostami, do których należy prototyp wykresu.
selectItems	query	Zwracaj właściwość `items` z pozycjami i prototypami pozycji używanymi w prototypie wykresu.
selectTemplateGroups	query	Zwracaj właściwość `templategroups` z grupami szablonów, do których należy prototyp wykresu.
selectTemplates	query	Zwracaj właściwość `templates` z szablonami, do których należy prototyp wykresu.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy prototyp wykresu; `hostid` - ID hosta, do którego należy prototyp wykresu.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `graphid`, `name`, `graphtype`, `discovered`.
countOutput	boolean	Te parametry są opisane w reference commentary.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie prototypów wykresów z reguły LLD

Pobierz wszystkie prototypy wykresów z reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.get",
    "params": {
        "output": "extend",
        "discoveryids": "27426"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "graphid": "1017",
            "name": "Wykorzystanie miejsca na dysku {#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
}
```

Zobacz także

- Reguła wykrywania
- Pozycja wykresu
- Pozycja
- Host
- Grupa hostów
- Szablon
- Grupa szablonów

Źródło

CGraphPrototype::get() w *ui/include/classes/api/services/CGraphPrototype.php*.

graphprototype.update

Opis

`object graphprototype.update(object/array graphPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów wykresów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu wykresu do zaktualizowania.

Właściwość `graphid` musi być zdefiniowana dla każdego prototypu wykresu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu wykresu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
gitems	array	Pozycje wykresu służące do zastąpienia istniejących pozycji wykresu. Jeśli pozycja wykresu ma zdefiniowaną właściwość `gitemid`, zostanie zaktualizowana; w przeciwnym razie zostanie utworzona nowa pozycja wykresu.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów wykresów we właściwości `graphids`.

Przykłady

Zmiana rozmiaru prototypu wykresu

Zmień rozmiar prototypu wykresu na 1100 na 400 pikseli.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graphprototype.update",
    "params": {
        "graphid": "439",
        "width": 1100,
        "height": 400
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Źródło

CGraphPrototype::update() w *ui/include/classes/api/services/CGraphPrototype.php*.

Prototyp wyzwalacza

Ta klasa jest przeznaczona do pracy z prototypami wyzwalaczy.

Odwołania do obiektów:

- Prototyp wyzwalacza
- Tag prototypu wyzwalacza

Dostępne metody:

- triggerprototype.create - tworzenie nowych prototypów wyzwalaczy
- triggerprototype.delete - usuwanie prototypów wyzwalaczy
- triggerprototype.get - pobieranie prototypów wyzwalaczy
- triggerprototype.update - aktualizowanie prototypów wyzwalaczy

Obiekt prototypu wyzwalacza

Następujące obiekty są bezpośrednio powiązane z API `triggerprototype`.

Prototyp wyzwalacza

Obiekt prototypu wyzwalacza ma następujące właściwości.

Właściwość	Typ	Opis
triggerid	ID	ID prototypu wyzwalacza. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	string	Nazwa prototypu wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
expression	string	Wyrażenie wyzwalacza. Musi zawierać co najmniej jeden prototyp pozycji. Prototypy pozycji z ustawionym `value_type` na 5 (binary) lub 6 (JSON) nie mogą być używane w wyrażeniach wyzwalaczy. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
event_name	string	Nazwa zdarzenia generowanego przez wyzwalacz.
opdata	string	Dane operacyjne.
comments	string	Dodatkowe komentarze do prototypu wyzwalacza.
priority	integer	Poziom ważności prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa.
status	integer	Czy prototyp wyzwalacza jest włączony czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
flags	integer	Pochodzenie prototypu wyzwalacza. Możliwe wartości: 2 - prototyp wyzwalacza; 6 - wykryty prototyp wyzwalacza. Zachowanie właściwości: - *tylko do odczytu*
templateid	ID	ID nadrzędnego prototypu wyzwalacza szablonu. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
type	integer	Czy prototyp wyzwalacza może generować wiele zdarzeń problemowych. Możliwe wartości: 0 - *(domyślnie)* nie generuj wielu zdarzeń; 1 - generuj wiele zdarzeń.
url	string	URL powiązany z prototypem wyzwalacza.
url_name	string	Etykieta dla URL powiązanego z prototypem wyzwalacza.
recovery_mode	integer	Tryb generowania zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* Wyrażenie; 1 - Wyrażenie odzyskiwania; 2 - Brak.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza. Musi zawierać co najmniej jeden prototyp pozycji. Prototypy pozycji z ustawionym `value_type` na 5 (binary) lub 6 (JSON) nie mogą być używane w wyrażeniach odzyskiwania wyzwalacza.
correlation_mode	integer	Zamykanie zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* Wszystkie problemy; 1 - Wszystkie problemy, jeśli wartości tagów są zgodne.
correlation_tag	string	Tag do dopasowania.
manual_close	integer	Zezwalaj na ręczne zamknięcie. Możliwe wartości: 0 - *(domyślnie)* Nie; 1 - Tak.
discover	integer	Status wykrywania prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* nowe wyzwalacze będą wykrywane; 1 - nowe wyzwalacze nie będą wykrywane, a istniejące wyzwalacze zostaną oznaczone jako utracone.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych prototypów wyzwalaczy z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli prototyp wyzwalacza należy do szablonu

Tag wyzwalacza prototypu

Obiekt tagu wyzwalacza prototypu ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu wyzwalacza prototypu. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu wyzwalacza prototypu. Zachowanie właściwości: - *tylko do odczytu*.

Właściwość	Typ	Opis
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon; 1 - host; 3 - pozycja; 4 - prototyp pozycji. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

triggerprototype.create

Opis

`object triggerprototype.create(object/array triggerPrototypes)`

Ta metoda umożliwia tworzenie nowych prototypów wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Prototypy wyzwalaczy do utworzenia.

Oprócz standardowych właściwości prototypu wyzwalacza metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dependencies	array	Wyzwalacze i prototypy wyzwalaczy, od których zależy prototyp wyzwalacza. Wyzwalacze muszą mieć zdefiniowaną tylko właściwość `triggerid`.
tags	array	Tagi prototypu wyzwalacza.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych prototypów wyzwalaczy we właściwości `triggerids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych prototypów wyzwalaczy.

Przykłady

Tworzenie prototypu wyzwalacza

Utwórz prototyp wyzwalacza, aby wykrywać, kiedy system plików ma mniej niż 20% wolnego miejsca na dysku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.create",
    "params": {
        "description": "Wolne miejsce na dysku jest mniejsze niż 20% na woluminie {#FSNAME}",
        "expression": "last(/Zabbix server/vfs.fs.size[{#FSNAME},pfree])<20",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
```

```
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17372"
        ]
    },
    "id": 1
}
```

Źródło

CTriggerPrototype::create() w *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.delete

Opis

`object triggerprototype.delete(array triggerPrototypeIds)`

Ta metoda umożliwia usuwanie prototypów wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID prototypów wyzwalaczy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych prototypów wyzwalaczy we właściwości `triggerids`.

Przykłady

Usuwanie wielu prototypów wyzwalaczy

Usuń dwa prototypy wyzwalaczy.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "triggerprototype.delete",
"params": [
"12002",
"12003"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
```

```
"triggerids": [
"12002",
"12003"
]
},
"id": 1
}
```

Źródło

CTriggerPrototype::delete() w *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.get

Opis

`integer/array triggerprototype.get(object parameters)`

Metoda umożliwia pobieranie prototypów wyzwalaczy zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
active	flag	Zwracaj tylko włączone prototypy wyzwalaczy należące do monitorowanych hostów.
discoveryids	ID/array	Zwracaj tylko prototypy wyzwalaczy należące do podanych reguł LLD.
functions	string/array	Zwracaj tylko wyzwalacze używające podanych funkcji. Listę obsługiwanych funkcji można znaleźć na stronie Supported functions.
group	string	Zwracaj tylko prototypy wyzwalaczy należące do hostów lub szablonów z grup hostów lub grup szablonów o podanej nazwie.
groupids	ID/array	Zwracaj tylko prototypy wyzwalaczy należące do hostów lub szablonów z podanych grup hostów lub grup szablonów.
host	string	Zwracaj tylko prototypy wyzwalaczy należące do hostów o podanej nazwie.
hostids	ID/array	Zwracaj tylko prototypy wyzwalaczy należące do podanych hostów.
inherited	boolean	Jeśli ustawiono na `true`, zwracaj tylko prototypy wyzwalaczy dziedziczone z szablonu.
maintenance	boolean	Jeśli ustawiono na `true`, zwracaj tylko włączone prototypy wyzwalaczy należące do hostów w konserwacji.
min_severity	integer	Zwracaj tylko prototypy wyzwalaczy o ważności większej lub równej podanej ważności.
monitored	flag	Zwracaj tylko włączone prototypy wyzwalaczy należące do monitorowanych hostów i zawierające wyłącznie włączone pozycje.
templated	boolean	Jeśli ustawiono na `true`, zwracaj tylko prototypy wyzwalaczy należące do szablonów.
templateids	ID/array	Zwracaj tylko prototypy wyzwalaczy należące do podanych szablonów.
triggerids	ID/array	Zwracaj tylko prototypy wyzwalaczy o podanych identyfikatorach.
expandExpression	flag	Rozwiń funkcje i makra w wyrażeniu wyzwalacza.
selectDependencies	query	Zwracaj prototypy wyzwalaczy i wyzwalacze, od których zależy prototyp wyzwalacza, we właściwości `dependencies`.

Parametr	Typ	Opis
selectDiscoveryData	query	Zwracaj właściwość `discoveryData` z danymi obiektu wykrywania prototypu wyzwalacza. Obiekt wykrywania prototypu wyzwalacza łączy wykryty prototyp wyzwalacza z prototypem wyzwalacza, z którego został wykryty. Ma on następujące właściwości: `parent_triggerid` - `(ID)` identyfikator prototypu wyzwalacza, z którego utworzono prototyp wyzwalacza; `status` - `(int)` status wykrywania prototypu wyzwalacza: 0 - *(domyślnie)* prototyp wyzwalacza jest wykryty, 1 - prototyp wyzwalacza nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, kiedy prototyp wyzwalacza, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, kiedy prototyp wyzwalacza, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik określający, czy prototyp wyzwalacza został wyłączony przez regułę LLD czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwracaj regułę LLD, do której należy prototyp wyzwalacza, we właściwości `discoveryRule`.
selectDiscoveryRulePrototype	query	Zwracaj właściwość `discoveryRulePrototype` z nadrzędnym prototypem reguły LLD, do którego należy prototyp wyzwalacza.
selectFunctions	query	Zwracaj funkcje używane w prototypie wyzwalacza we właściwości `functions`. Obiekty funkcji reprezentują funkcje używane w wyrażeniu wyzwalacza i mają następujące właściwości: `functionid` - `(ID)` identyfikator funkcji; `itemid` - `(ID)` identyfikator pozycji użytej w funkcji; `function` - `(string)` nazwa funkcji; `parameter` - `(string)` parametr przekazany do funkcji. Parametr zapytania jest zastępowany symbolem $ w zwracanym ciągu.
selectHostGroups	query	Zwracaj grupy hostów, do których należy prototyp wyzwalacza, we właściwości `hostgroups`.
selectHosts	query	Zwracaj hosty, do których należy prototyp wyzwalacza, we właściwości `hosts`.
selectInheritedTags	query	Zwracaj właściwość `inheritedTags` z tagami dziedziczonymi z szablonów, podłączonych szablonów i hostów oraz z prototypów pozycji, do których odwołują się wyrażenia wyzwalacza lub odtwarzania.
selectItems	query	Zwracaj pozycje i prototypy pozycji używane przez prototyp wyzwalacza we właściwości `items`.
selectTags	query	Zwracaj tagi prototypu wyzwalacza we właściwości `tags`.
selectTemplateGroups	query	Zwracaj grupy szablonów, do których należy prototyp wyzwalacza, we właściwości `templategroups`.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy prototyp wyzwalacza; `hostid` - identyfikator hosta, do którego należy prototyp wyzwalacza.

Parametr	Typ	Opis
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wyniki będą sortowane według `host`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `triggerid`, `description`, `status`, `priority`, `discovered`.
countOutput	boolean	Te parametry opisano w reference commentary.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie prototypów wyzwalaczy z reguły LLD

Pobierz wszystkie prototypy wyzwalaczy i ich funkcje z reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": "extend",
        "selectFunctions": "extend",
        "discoveryids": "22450"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "13272",
            "expression": "{12598}<20",
            "description": "Liczba wolnych i-węzłów jest mniejsza niż 20% na woluminie {#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": "$",
                    "function": "last"
                }
            ]
        },
        {
            "triggerid": "13266",
            "expression": "{13500}<20",
            "description": "Wolne miejsce na dysku jest mniejsze niż 20% na woluminie {#FSNAME}",
            "url": "",
            "status": "0",
            "value": "0",
            "priority": "2",
            "lastchange": "0",
            "comments": "",
            "error": "",
            "templateid": "0",
            "type": "0",
            "state": "0",
            "flags": "2",
            "recovery_mode": "0",
            "recovery_expression": "",
            "correlation_mode": "0",
            "correlation_tag": "",
            "manual_close": "0",
            "opdata": "",
            "discover": "0",
            "event_name": "",
            "uuid": "74a1fc62bfe24b7eabe4e244c70dc384",
            "url_name": "",
            "functions": [
                {
                    "functionid": "13500",
                    "itemid": "22686",
                    "triggerid": "13266",
                    "parameter": "$",
                    "function": "last"
                }
            ]
        }
    ],
    "id": 1
}
```


Pobieranie określonego prototypu wyzwalacza z tagami

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "17373",
            "description": "Wolne miejsce na dysku jest mniejsze niż 20% na woluminie {#FSNAME}",
            "tags": [
                {
                    "tag": "volume",
                    "value": "{#FSNAME}"
                },
                {
                    "tag": "type",
                    "value": "{#FSTYPE}"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Reguła wykrywania
- Pozycja
- Host
- Grupa hostów
- Grupa szablonów

Źródło

CTriggerPrototype::get() w *ui/include/classes/api/services/CTriggerPrototype.php*.

triggerprototype.update

Opis

`object triggerprototype.update(object/array triggerPrototypes)`

Ta metoda umożliwia aktualizację istniejących prototypów wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości prototypu wyzwalacza do zaktualizowania.

Właściwość `triggerid` musi być zdefiniowana dla każdego prototypu wyzwalacza, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości prototypu wyzwalacza metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dependencies	array	Wyzwalacze i prototypy wyzwalaczy, od których zależy prototyp wyzwalacza. Dla wyzwalaczy musi być zdefiniowana tylko właściwość `triggerid`.
tags	array	Tagi prototypu wyzwalacza.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych prototypów wyzwalaczy we właściwości `triggerids`.

Przykłady

Włączanie prototypu wyzwalacza

Włącz prototyp wyzwalacza, to znaczy ustaw jego status na ”0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Zastępowanie tagów prototypu wyzwalacza

Zastąp tagi dla jednego prototypu wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "triggerprototype.update",
    "params": {
        "triggerid": "17373",
        "tags": [
            {
                "tag": "volume",
                "value": "{#FSNAME}"
            },
            {
                "tag": "type",
                "value": "{#FSTYPE}"
            }
        ]
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}
```

Źródło

CTriggerPrototype::update() w *ui/include/classes/api/services/CTriggerPrototype.php*.

Proxy

Ta klasa jest przeznaczona do pracy z proxy.

Odwołania do obiektów:

- Proxy

Dostępne metody:

- proxy.create - tworzenie nowych proxy
- proxy.delete - usuwanie proxy
- proxy.get - pobieranie proxy
- proxy.update - aktualizacja proxy

Obiekt proxy

Następujące obiekty są bezpośrednio powiązane z API `proxy`.

Proxy

Obiekt proxy ma następujące właściwości.

Property	Type	Description
proxyid	ID	Identyfikator proxy. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
name	string	Nazwa proxy. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
proxy_groupid	ID	Identyfikator grupy proxy. 0 - proxy nie jest przypisane do żadnej grupy proxy.
local_address	string	Adres dla aktywnych agentów. Adres IP lub nazwa DNS, z którym należy się połączyć. Zachowanie właściwości: - *wymagane*, jeśli `proxy_groupid` nie ma wartości 0

Property	Type	Description
local_port	string	Numer lokalnego portu proxy, z którym należy się połączyć. Obsługiwane są makra użytkownika. Wartość domyślna: 10051. Zachowanie właściwości: - *obsługiwane*, jeśli `proxy_groupid` nie ma wartości 0
operating_mode	integer	Typ proxy. Możliwe wartości: 0 - aktywne proxy; 1 - pasywne proxy. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
description	text	Opis proxy.
lastaccess	timestamp	Czas ostatniego połączenia proxy z serwerem. Zachowanie właściwości: - *tylko do odczytu*
address	string	Adres IP lub nazwa DNS, z którym należy się połączyć. Obsługiwane są makra użytkownika. Zachowanie właściwości: - *wymagane*, jeśli tryb działania proxy Zabbix jest pasywny
port	string	Numer portu, z którym należy się połączyć. Obsługiwane są makra użytkownika. Wartość domyślna: 10051. Zachowanie właściwości: - *obsługiwane*, jeśli tryb działania proxy Zabbix jest pasywny
allowed_addresses	string	Adresy IP lub nazwy DNS aktywnego proxy Zabbix rozdzielone przecinkami.
tls_connect	integer	Połączenia z proxy. Możliwe wartości: 1 - *(domyślnie)* brak szyfrowania; 2 - PSK; 4 - certyfikat.
tls_accept	integer	Połączenia z proxy. Możliwe wartości bitmapy: 1 - *(domyślnie)* brak szyfrowania; 2 - PSK; 4 - certyfikat. To pole jest maską bitową; dopuszczalna jest dowolna suma możliwych wartości bitmapy (na przykład 6 dla PSK i certyfikatu).
tls_issuer	string	Wystawca certyfikatu.
tls_subject	string	Podmiot certyfikatu.

Property	Type	Description
tls_psk_identity	string	Tożsamość PSK; musi być powiązana tylko z jednym kluczem PSK (w ramach automatycznej rejestracji, hostów i proxy). Nie umieszczaj poufnych informacji w tożsamości PSK, ponieważ jest ona przesyłana przez sieć w postaci niezaszyfrowanej, aby poinformować odbiorcę, którego klucza PSK należy użyć. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `tls_connect` ma wartość „PSK” lub `tls_accept` zawiera bit „PSK”
tls_psk	string	Klucz współdzielony (PSK); musi zawierać co najmniej 32 cyfry szesnastkowe. Zachowanie właściwości: - *tylko do zapisu* - *wymagane*, jeśli `tls_connect` ma wartość „PSK” lub `tls_accept` zawiera bit „PSK”
custom_timeouts	integer	Określa, czy zastąpić globalne limity czasu pozycji na poziomie proxy. Możliwe wartości: 0 - *(domyślnie)* użyj ustawień globalnych; 1 - zastąp limity czasu.
timeout_zabbix_agent	string	Przetwarzanie nie może trwać dłużej niż `timeout_*` sekund. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m). Akceptuje również makra użytkownika. Używane tylko wtedy, gdy `custom_timeouts` ma wartość „override timeouts”; w przeciwnym razie wartość jest ignorowana. Możliwe wartości mieszczą się w zakresie: 1-600 s. Zachowanie właściwości: - *wymagane*, jeśli `custom_timeouts` ma wartość „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		
version	integer	Wersja proxy. Trzyczęściowy numer wersji Zabbix, w którym każda część jest zapisana za pomocą dwóch cyfr dziesiętnych, np. 60401 dla wersji 6.4.1, 70002 dla wersji 7.0.2 itd. 0 - Nieznana wersja proxy. Zachowanie właściwości: - *tylko do odczytu*

Property	Type	Description
compatibility	integer	Wersja proxy w porównaniu z wersją serwera Zabbix. Możliwe wartości: 0 - Niezdefiniowana; 1 - Bieżąca wersja (proxy i serwer mają tę samą wersję główną); 2 - Nieaktualna wersja (wersja proxy jest starsza niż wersja serwera, ale jest częściowo obsługiwana); 3 - Nieobsługiwana wersja (wersja proxy jest starsza niż poprzednia wersja LTS serwera lub główna wersja serwera jest starsza niż główna wersja proxy). Zachowanie właściwości: - *tylko do odczytu*
state	integer	Stan proxy. Możliwe wartości: 0 - Nieznany; 1 - Offline; 2 - Online. Zachowanie właściwości: - *tylko do odczytu*

Monitorowanie wydajności aplikacji (APM)

Obiekt APM ma następujące właściwości.

Property	Type	Description
data_collection_status	integer	Określa, czy zbieranie danych telemetrycznych jest włączone na proxy. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone.
max_messages_per_second	integer	Maksymalna liczba komunikatów na sekundę akceptowanych przez proxy. Zakres możliwych wartości: 0 (bez ograniczeń) – 2147483647. Wartość domyślna: 0. Zachowanie właściwości: - *obsługiwana*, jeśli `data_collection_status` jest ustawione na „Włączone”
additional_resource_attributes	array	Tablica atrybutów zasobów dodawanych do przychodzących danych telemetrycznych. Jeśli przychodzące dane zawierają już atrybut zasobu o tej samej nazwie, zachowywany jest atrybut przychodzący zamiast atrybutu określonego tutaj. Zachowanie właściwości: - *obsługiwana*, jeśli `data_collection_status` jest ustawione na „Włączone”

Atrybuty zasobu

Obiekt atrybutów zasobu ma następujące właściwości.

Property	Type	Description
key	string	Nazwa atrybutu zasobu. Musi być unikatowa dla określonego `signal_type`. Zachowanie właściwości: - *wymagane*
value	string	Wartość atrybutu zasobu. Może być pusta.
signal_type	integer	Typ danych telemetrycznych, do którego odnosi się atrybut. Możliwe wartości: 0 - Ślady; 1 - Metryki; 2 - Logi. Zachowanie właściwości: - *wymagane*

proxy.create

Opis

`object proxy.create(object/array proxies)`

Ta metoda umożliwia tworzenie nowych proxy.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienie do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Proxy do utworzenia.

Oprócz standardowych właściwości proxy metoda przyjmuje następujące parametry.

Parametr	Typ	Opis
hosts	array	Hosty, które mają być monitorowane przez proxy. Jeśli host jest już monitorowany przez inny proxy, zostanie przypisany do bieżącego proxy. Hosty muszą mieć zdefiniowaną wyłącznie właściwość `hostid`.
apm	object	Ustawienia monitorowania wydajności aplikacji (APM) proxy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych proxy we właściwości `proxyids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych proxy.

Przykłady

Utwórz aktywny proxy

Utwórz aktywny proxy „Active proxy” i przypisz do niego host do monitorowania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Active proxy",
        "operating_mode": "0",
        "hosts": [
```

```
            {
                "hostid": "10279"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10280"
        ]
    },
    "id": 1
}
```

Utwórz pasywny proxy

Utwórz pasywny proxy „Passive proxy” i przypisz do niego dwa hosty do monitorowania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "name": "Passive proxy",
        "operating_mode": "1",
        "address": "127.0.0.1",
        "port": "10051",
        "hosts": [
            {
                "hostid": "10192"
            },
            {
                "hostid": "10139"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10284"
        ]
    },
    "id": 1
}
```

Tworzenie proxy i dodawanie go do grupy proxy

Utwórz aktywny proxy „Active proxy” i dodaj go do grupy proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
```

```
        "name": "Active proxy",
        "proxy_groupid": "1",
        "operating_mode": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Tworzenie proxy z szyfrowaniem opartym na certyfikatach

Utwórz aktywny proxy z szyfrowaniem opartym na certyfikatach szyfrowanie. Proxy jest skonfigurowany do monitorowania hosta "10785".

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "2"
        ]
    },
    "id": 1
}
```

Tworzenie proxy z włączoną funkcją APM

Utwórz aktywne proxy „Active proxy" z włączoną funkcją APM.
Proxy akceptuje do 1000 komunikatów na sekundę, dodaje atrybut zasobu `deployment.environment` do śladów oraz atrybut zasobu `cloud.region` do metryk.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host
- Grupa proxy

Źródło

CProxy::create() w *ui/include/classes/api/services/CProxy.php*.

proxy.delete

Opis

`object proxy.delete(array proxies)`

Ta metoda umożliwia usuwanie proxy.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID proxy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych proxy we właściwości `proxyids`.

Przykłady

Usuń wiele proxy

Usuń dwa proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.delete",
    "params": [
        "10286",
        "10285"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10286",
            "10285"
        ]
    },
    "id": 1
}
```

Źródło

CProxy::delete() w *ui/include/classes/api/services/CProxy.php*.

proxy.get

Opis

`integer/array proxy.get(object parameters)`

Metoda umożliwia pobieranie proxy zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.
 Zwracane są tylko proxy dozwolone przez *listę dostępu proxy* grup użytkownika.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
proxyids	ID/array	Zwraca tylko proxy o podanych identyfikatorach.
proxy_groupids	ID/array	Zwraca tylko proxy należące do podanych grup proxy.
selectApm	query	Zwraca właściwość `apm` zawierającą ustawienia monitorowania wydajności aplikacji (APM) proxy.
selectAssignedHosts	query	Zwraca właściwość `assignedHosts` zawierającą hosty przypisane do proxy. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` zawierającą hosty monitorowane przez proxy. Obsługuje `count`.
selectProxyGroup	query	Zwraca właściwość `proxyGroup` zawierającą obiekt grupy proxy.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `proxyid`, `name`, `operating_mode`.
countOutput	boolean	Parametry te opisano w komentarzu referencyjnym.

Parameter	Type	Description
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich proxy

Pobierz wszystkie skonfigurowane proxy i ich interfejsy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Host
- Grupa proxy

Źródło

CProxy::get() w *ui/include/classes/api/services/CProxy.php*.

proxy.update

Opis

`object proxy.update(object/array proxies)`

Ta metoda umożliwia aktualizację istniejących proxy.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości proxy do zaktualizowania.

Właściwość `proxyid` musi być zdefiniowana dla każdego proxy, natomiast wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, a wszystkie pozostałe pozostaną niezmienione.

Oprócz standardowych właściwości proxy metoda akceptuje następujące parametry.

Parameter	Type	Description
hosts	array	Hosty, które mają być monitorowane przez proxy. Jeśli host jest już monitorowany przez inne proxy, zostanie przypisany do bieżącego proxy. Hosty muszą mieć zdefiniowaną wyłącznie właściwość `hostid`.
apm	object	Ustawienia monitorowania wydajności aplikacji (APM) proxy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych proxy we właściwości `proxyids`.

Przykłady

Zmień hosty monitorowane przez proxy

Zaktualizuj proxy, aby monitorował dwa podane hosty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "hosts": [
            {
                "hostid": "10294"
            },
            {
                "hostid": "10295"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
    },
    "id": 1
}
```

Zmień status proxy

Zmień proxy na aktywny proxy i zmień jego nazwę na „Active proxy”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.update",
    "params": {
        "proxyid": "10293",
        "name": "Active proxy",
        "operating_mode": "0"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "10293"
        ]
    },
    "id": 1
}
```

Dodawanie proxy do grupy proxy

Zaktualizuj proxy o ID „5” i dodaj go do grupy proxy o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "proxy.create",
    "params": {
        "proxyid": "5",
        "proxy_groupid": "1",
        "local_address": "127.0.0.1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "proxyids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Host
- Grupa proxy

Źródło

CProxy::update() w *ui/include/classes/api/services/CProxy.php*.

Pulpit

Ta klasa jest przeznaczona do pracy z pulpitami.

Odwołania do obiektów:

- Pulpit
- Strona pulpitu
  - Widżet pulpitu
    - Pole widżetu pulpitu
- Grupa użytkowników pulpitu
- Użytkownik pulpitu

Dostępne metody:

- dashboard.create - tworzenie nowych pulpitów

- dashboard.delete - usuwanie pulpitów
- dashboard.get - pobieranie pulpitów
- dashboard.update - aktualizowanie pulpitów

Obiekt Pulpitu Nawigacyjnego

Następujące obiekty są bezpośrednio związane z API `dashboard`.

Dashboard

Obiekt dashboard ma następujące właściwości.

Property	Type	Description
dashboardid	ID	ID dashboardu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa dashboardu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
userid	ID	ID użytkownika będącego właścicielem dashboardu.
private	integer	Typ udostępniania dashboardu. Możliwe wartości: 0 - publiczny dashboard; 1 - *(domyślnie)* prywatny dashboard.
display_period	integer	Domyślny interwał pokazu slajdów (w sekundach). Możliwe wartości: 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 30.
auto_start	integer	Automatyczne uruchamianie pokazu slajdów. Możliwe wartości: 0 - nie uruchamiaj automatycznie pokazu slajdów; 1 - *(domyślnie)* uruchamiaj automatycznie pokaz slajdów.

Strona dashboardu

Obiekt strony dashboardu ma następujące właściwości.

Property	Type	Description
dashboard_pageid	ID	ID strony dashboardu. Zachowanie właściwości: - *tylko do odczytu*
name	string	Nazwa strony dashboardu. Domyślnie: pusty ciąg znaków.
display_period	integer	Interwał pokazu slajdów dashboardu (w sekundach). Możliwe wartości: 0, 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 0 (zostanie użyty domyślny interwał pokazu slajdów).
widgets	array	Tablica obiektów widgetu dashboardu.

Widżet pulpitu

Obiekt widżetu pulpitu ma następujące właściwości.

Właściwość	Typ	Opis
widgetid	ID	ID widżetu pulpitu. Zachowanie właściwości: - *tylko do odczytu*
type	string	Typ widżetu pulpitu. Możliwe wartości: actionlog - Dziennik działań; clock - Zegar; discovery - Status wykrywania; favgraphs - Ulubione wykresy; favmaps - Ulubione mapy; gauge - Wskaźnik; geomap - Mapa geograficzna; graph - Wykres (klasyczny); graphprototype - Prototyp wykresu; honeycomb - Plaster miodu; hostavail - Dostępność hosta; hostcard - Karta hosta; hostnavigator - Nawigator hostów; itemcard - Karta pozycji; itemhistory - Historia pozycji; itemnavigator - Nawigator pozycji; item - Wartość pozycji; map - Mapa; navtree - Drzewo nawigacji mapy; piechart - Wykres kołowy; problemhosts - Hosty z problemami; problems - Problemy; problemsbysv - Problemy według ważności; scatterplot - Wykres punktowy; slareport - Raport SLA; svggraph - Wykres; systeminfo - Informacje o systemie; tophosts - Najlepsze hosty; topitems - Najlepsze pozycje; toptriggers - Najważniejsze wyzwalacze; trigover - Przegląd wyzwalaczy; url - URL; web - Monitorowanie WWW. Zachowanie właściwości: - *wymagane*
name	string	Niestandardowa nazwa widżetu.
x	integer	Pozycja pozioma od lewej strony pulpitu. Możliwe wartości mieszczą się w zakresie od 0 do 71.
y	integer	Pozycja pionowa od górnej krawędzi pulpitu. Możliwe wartości mieszczą się w zakresie od 0 do 63.
width	integer	Szerokość widżetu. Możliwe wartości mieszczą się w zakresie od 1 do 72.
height	integer	Wysokość widżetu. Możliwe wartości mieszczą się w zakresie od 1 do 64.
view_mode	integer	Tryb widoku widżetu. Możliwe wartości: 0 - *(domyślnie)* domyślny widok widżetu; 1 - z ukrytym nagłówkiem;

Właściwość	Typ	Opis
fields	array	Tablica obiektów pola widżetu pulpitu. Zachowanie właściwości: - patrz poszczególne widżety w Pola widżetu pulpitu

Pole widżetu pulpitu

Obiekt pola widżetu pulpitu ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ pola widżetu. Możliwe wartości: 0 - Liczba całkowita; 1 - Ciąg znaków; 2 - Grupa hostów; 3 - Host; 4 - Pozycja; 5 - Prototyp pozycji; 6 - Wykres; 7 - Prototyp wykresu; 8 - Mapa; 9 - Usługa; 10 - SLA; 11 - Użytkownik; 12 - Akcja; 13 - Typ nośnika. Zachowanie właściwości: - *wymagane*
name	string	Nazwa pola widżetu. Możliwe wartości: patrz Pola widżetu pulpitu. Zachowanie właściwości: - *wymagane*
value	mixed	Wartość pola widżetu zależna od typu. Możliwe wartości: patrz Pola widżetu pulpitu. Zachowanie właściwości: - *wymagane*

Grupa użytkowników pulpitu

Lista uprawnień pulpitu oparta na grupach użytkowników. Ma następujące właściwości.

Właściwość	Typ	Opis
usrgrpid	ID	ID grupy użytkowników. Zachowanie właściwości: - *wymagane*

Właściwość	Typ	Opis
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Zachowanie właściwości: - *wymagane*

Użytkownik pulpitu

Lista uprawnień pulpitu na podstawie użytkowników. Ma następujące właściwości.

Właściwość	Typ	Opis
userid	ID	ID użytkownika. Zachowanie właściwości: - *wymagane*
permission	integer	Typ poziomu uprawnień. Możliwe wartości: 2 - tylko do odczytu; 3 - odczyt-zapis. Zachowanie właściwości: - *wymagane*

dashboard.create

Opis

`object dashboard.create(object/array dashboards)`

Ta metoda umożliwia tworzenie nowych dashboardów.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Pulpity do utworzenia.

Oprócz standardowych właściwości pulpitu, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
pages	array	Strony pulpitu, które mają zostać utworzone dla pulpitu. Strony pulpitu będą uporządkowane w tej samej kolejności, w jakiej zostały podane. Zachowanie parametrów: - *wymagane*
users	array	Udziały użytkownika pulpitu, które mają zostać utworzone na pulpicie.
userGroups	array	Udziały grupy użytkowników pulpitu, które mają zostać utworzone na pulpicie.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych pulpitów we właściwości `dashboardids`. Kolejność zwróconych ID jest zgodna z kojenością przekazanych pulpitów.

Przykłady

Tworzenie pulpitu

Utwórz pulpit o nazwie „My dashboard” z jednym widżetem Problemy z tagami oraz z użyciem dwóch typów udostępniania (grupa użytkowników i użytkownik) na jednej stronie pulpitu.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Strona pulpitu
- Widżet pulpitu
- Pole widżetu pulpitu
- Użytkownik pulpitu
- Grupa użytkowników pulpitu

Źródło

CDashboard::create() w *ui/include/classes/api/services/CDashboard.php*.

dashboard.delete

Opis

`object dashboard.delete(array dashboardids)`

Ta metoda umożliwia usuwanie dashboardów.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` identyfikatory pulpitów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych pulpitów we właściwości `dashboardids`.

Przykłady

Usuwanie wielu pulpitów

Usuń dwa pulpity.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "dashboard.delete",
"params": [
"2",
"3"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"2",
"3"
]
},
```

```
"id": 1
}
```

Źródło

CDashboard::delete() w *ui/include/classes/api/services/CDashboard.php*.

dashboard.get

Opis

`integer/array dashboard.get(object parameters)`

Ta metoda umożliwia pobieranie dashboardów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dashboardids	ID/array	Zwraca tylko pulpity z podanymi ID.
selectPages	query	Zwraca właściwość `pages` z prawidłowo uporządkowanymi stronami pulpitu.
selectUsers	query	Zwraca właściwość `users` z użytkownikami, którym udostępniono pulpit.
selectUserGroups	query	Zwraca właściwość `userGroups` z grupami użytkowników, którym udostępniono pulpit.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dashboardid`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablica obiektów;
- ilość pobranych obiektów, jeżeli parametr `countOutput` został użyty.

Przykłady

Pobieranie pulpitu nawigacyjnego według ID

Pobierz wszystkie dane o pulpitach nawigacyjnych „1” i „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "dashboardids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Strona pulpitu
- Widżet pulpitu
- Pole widżetu pulpitu
- Użytkownik pulpitu
- Grupa użytkowników pulpitu

Źródło

CDashboard::get() w *ui/include/classes/api/services/CDashboard.php*.

dashboard.update

Opis

`object dashboard.update(object/array dashboards)`

Ta metoda umożliwia aktualizację istniejących dashboardów.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości dashboardu do zaktualizowania.

Właściwość `dashboardid` musi być zdefiniowana dla każdego dashboardu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości dashboardu, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
pages	array	Strony dashboardu, które zastąpią istniejące strony dashboardu. Strony dashboardu są aktualizowane za pomocą właściwości `dashboard_pageid`. Nowe strony dashboardu zostaną utworzone dla obiektów bez właściwości `dashboard_pageid`, a istniejące strony dashboardu zostaną usunięte, jeśli nie zostaną ponownie użyte. Strony dashboardu zostaną uporządkowane w tej samej kolejności, w jakiej zostały podane. Zaktualizowane zostaną tylko określone właściwości stron dashboardu.
users	array	Udostępnienia użytkownikom dashboardu, które zastąpią istniejące elementy.
userGroups	array	Udostępnienia grupom użytkowników dashboardu, które zastąpią istniejące elementy.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych pulpitów pod właściwością `dashboardids`.

Przykłady

Zmienianie nazwy pulpitu

Zmień nazwę pulpitu na „SQL server status”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
```

```
        "dashboardid": "2",
        "name": "SQL server status"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Aktualizowanie stron pulpitu

Zmień nazwę pierwszej strony pulpitu, zastąp widgety na drugiej stronie pulpitu i dodaj nową stronę jako trzecią. Usuń wszystkie pozostałe strony pulpitu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "pages": [
            {
                "dashboard_pageid": 1,
                "name": "Strona o zmienionej nazwie"
            },
            {
                "dashboard_pageid": 2,
                "widgets": [
                    {
                        "type": "clock",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3
                    }
                ]
            },
            {
                "display_period": 60
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zmień właściciela pulpitu

Dostępne tylko dla administratorów i super administratorów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.update",
    "params": {
        "dashboardid": "2",
        "userid": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Strona pulpitu
- Widżet pulpitu
- Pole widżetu pulpitu
- Użytkownik pulpitu
- Grupa użytkowników pulpitu

Źródło

CDashboard::update() w *ui/include/classes/api/services/CDashboard.php*.

Pola widżetu pulpitu

Ta strona zawiera linki nawigacyjne do parametrów widgetów pulpitu oraz możliwych wartości właściwości dla odpowiednich obiektów pola widgetu pulpitu.

Aby zobaczyć parametry i wartości właściwości dla każdego widgetu, użyj paska bocznego, aby przejść do strony odpowiedniego widgetu.

1 Log akcji

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Dziennik działań* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Dziennik działań*, zapoznaj się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widżet *Action log*.

Parameter	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Odbiorcy	11	userids.0	Identyfikator użytkownika User. Uwaga: Aby skonfigurować wielu użytkowników, utwórz obiekt pola widżetu dashboardu dla każdego użytkownika, zwiększając numer w nazwie właściwości.
Działania	12	actionids.0	Identyfikator Action. Uwaga: Aby skonfigurować wiele działań, utwórz obiekt pola widżetu dashboardu dla każdego działania, zwiększając numer w nazwie właściwości.
Typy mediów	13	mediatypeids.0	Identyfikator Media type. Uwaga: Aby skonfigurować wiele typów mediów, utwórz obiekt pola widżetu dashboardu dla każdego typu mediów, zwiększając numer w nazwie właściwości.
Stan	0	statuses.0	0 - W toku; 1 - Wysłano/Wykonano; 2 - Niepowodzenie. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Ciąg wyszukiwania	1	message	Dowolna wartość ciągu znaków.
Okres	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor okresu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie możesz ustawić okres wyłącznie za pomocą parametrów *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli parametr *Okres* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.to`
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli parametr *Okres* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.from`
Sortuj wpisy według	0	sort_triggers	3 - Czas (rosnąco); 4 - *(domyślnie)* Czas (malejąco); 5 - Typ (rosnąco); 6 - Typ (malejąco); 7 - Stan (rosnąco); 8 - Stan (malejąco); 11 - Odbiorca (rosnąco); 12 - Odbiorca (malejąco).

Parameter	type	name	value
Pokaż wiersze	0	show_lines	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 25.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Zegara*. Aby uzyskać więcej informacji na temat konfigurowania pulpitu nawigacyjnego, zobacz `dashboard.create`.

Konfigurowanie widżetu *Dziennik działań*

Skonfiguruj widżet *Dziennik działań*, który wyświetla 10 wpisów ze szczegółami operacji działania, posortowanych według czasu (rosnąco). Dodatkowo wyświetlaj szczegóły tylko dla tych operacji działania, które próbowały wysłać wiadomość e-mail do użytkownika „1”, ale zakończyły się niepowodzeniem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "actionlog",
                        "name": "Action log",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

2 Zegar

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Zegar* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Zegar*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Clock*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.

Parametr	type	name	value
Typ czasu	0	time_type	0 - *(domyślnie)* Czas lokalny; 1 - Czas serwera; 2 - Czas hosta.
Typ zegara	0	clock_type	0 - *(domyślnie)* Analogowy; 1 - Cyfrowy.

Następujące parametry są obsługiwane, jeśli *Typ czasu* jest ustawiony na „Czas hosta”.

Parametr	type	name	value
Pozycja	4	itemid.0	ID pozycji. Zachowanie parametru: - *wymagane*

Następujące parametry są obsługiwane, jeśli *Typ zegara* jest ustawiony na „Cyfrowy”.

Parametr	type	name	value
Pokaż	0	show.0	1 - Data; 2 - *(domyślnie)* Czas; 3 - Strefa czasowa. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z inkrementowanym numerem w nazwie właściwości.

Konfiguracja zaawansowana

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Typ zegara* jest ustawiony na „Cyfrowy”.

Parametr	type	name	value
Kolor tła	1	bg_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: "" (puste).

Data

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Clock type* jest ustawiony na „Digital”, a *Show* jest ustawione na „Date”.

Parametr	type	name	value
Pogrubienie	0	date_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	date_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: "" (puste).

Czas

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli *Typ zegara* jest ustawiony na „Cyfrowy”, a *Pokaż* jest ustawiony na „Czas”.

Parametr	typ	nazwa	wartość
Pogrubienie	0	time_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Parametr	typ	nazwa	wartość
Kolor	1	time_color	Kod koloru szesnastkowego (np. FF0000). Domyślnie: "" (puste).
Sekundy	0	time_sec	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Format	0	time_format	0 - *(domyślny)* 24-godzinny; 1 - 12-godzinny.

Strefa czasowa

Następujące zaawansowane parametry konfiguracyjne są obsługiwane, jeśli *Clock type* jest ustawiony na „Digital”, a *Show* jest ustawione na „Time zone”.

Parametr	type	name	value
Pogrubienie	0	tzone_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	tzone_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Strefa czasowa	1	tzone_timezone	Prawidłowy ciąg strefy czasowej (np. Europe/Riga, system, UTC itd.). Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP. Domyślnie: local. Zachowanie parametru: - *obsługiwany*, jeśli *Time type* jest ustawiony na „Local time” lub „Server time”
Format	0	tzone_format	0 - *(domyślnie)* Krótki; 1 - Pełny. Zachowanie parametru: - *obsługiwany*, jeśli *Time type* jest ustawiony na „Local time” lub „Server time”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Zegara*. Aby uzyskać więcej informacji na temat konfigurowania pulpitu nawigacyjnego, zobacz `dashboard.create`.

Konfigurowanie widżetu *Clock*

Skonfiguruj widżet *Clock*, który wyświetla lokalną datę, godzinę i strefę czasową w dostosowanym zegarze cyfrowym.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "clock",
                        "name": "Clock",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

3 Stan wykrywania

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widżetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widżetu *Stan wykrywania* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Status wykrywania*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Status wykrywania*. Więcej informacji na temat konfiguracji pulpitu znajduje się w `dashboard.create`.

Konfigurowanie widżetu *Discovery status*

Skonfiguruj widżet *Discovery status* z interwałem odświeżania ustawionym na 15 minut.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
```

```
                "widgets": [
                    {
                        "type": "discovery",
                        "name": "Discovery status",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

4 Ulubione wykresy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widżetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widżetu *Ulubione wykresy* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Favorite graphs*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Ulubione wykresy*. Więcej informacji na temat konfigurowania pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Ulubione wykresy*

Skonfiguruj widżet *Ulubione wykresy* z interwałem odświeżania ustawionym na 10 minut.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "Mój pulpit",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "favgraphs",
                        "name": "Ulubione wykresy",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

5 Ulubione mapy

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widżetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widżetu *Ulubione mapy* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Ulubione mapy*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Ulubione mapy*. Więcej informacji na temat konfiguracji pulpitu znajduje się w `dashboard.create`.

Konfigurowanie widżetu *Ulubione mapy*

Skonfiguruj widżet *Ulubione mapy* z interwałem odświeżania ustawionym na 10 minut.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

6 Wskaźnik

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Gauge* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Gauge*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Gauge*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Pozycja	4	itemid.0	ID pozycji. Zachowanie parametrów: - *wymagane*, jeśli *Pozycja (widżet)* nie jest ustawiona
Pozycja (widżet)	1	itemid._reference	Zamiast ID pozycji: `ABCDE._itemid` - ustawia kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla pozycji. Zachowanie parametrów: - *wymagane*, jeśli *Pozycja* nie jest ustawiona
Min	1	min	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. "1d", "2w", "4K", "8G"). Domyślnie: "0".
Max	1	max	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. "1d", "2w", "4K", "8G"). Domyślnie: "100".
Łuk wartości	1	value_arc_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: "" (puste).
Tło łuku	1	empty_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: "" (puste).
Tło	1	bg_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: "" (puste).
Pokaż	0	show.0	1 - Opis; 2 - Wartość; 3 - Wskazówka; 4 - Skala; 5 - Łuk wartości. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z inkrementowanym numerem w nazwie właściwości. Domyślnie: 1, 2, 4, 5. Wartości „Wskazówka" i „Skala" nie są obsługiwane, jeśli jednocześnie: - nie ustawiono obiektu pola widżetu pulpitu dla *Pokaż* z wartością „Łuk wartości"; - parametr konfiguracji zaawansowanej *Pokaż łuk* jest ustawiony na „Wyłączone". Parametry konfiguracji zaawansowanej dla opcji *Pokaż* nie są obsługiwane, jeśli nie ustawiono obiektów pól widżetu pulpitu z odpowiednimi wartościami.

Parametr	type	name	value
Nadpisz host	1	override_hostid._reference	`ABCDE._hostid` - ustawia kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustawia selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Zaawansowana konfiguracja

Następujące zaawansowane parametry konfiguracji są obsługiwane przez widżet *Gauge*.

Note:
Liczba w nazwie właściwości *Thresholds* (np. thresholds.0.color) odwołuje się do miejsca progu na liście posortowanej rosnąco. Jeśli jednak progi zostaną skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontend Zabbix (np. `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parameter	type	name	value
Kąt	0	angle	Możliwe wartości: 180 *(domyślnie)* lub 270.
Opis			
Opis	1	description	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, makra użytkownika. Domyślnie: `{ITEM.NAME}`.
Rozmiar	0	desc_size	Możliwe wartości od 1 do 100. Domyślnie: 15.
Pozycja w pionie	0	desc_v_pos	0 - U góry; 1 - *(domyślnie)* Na dole.
Pogrubienie	0	desc_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	desc_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Wartość			
Miejsca dziesiętne	0	decimal_places	Możliwe wartości od 1 do 10. Domyślnie: 2.
Rozmiar	0	value_size	Możliwe wartości od 1 do 100. Domyślnie: 25.
Pogrubienie	0	value_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	value_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).
Jednostki			
Jednostki (pole wyboru)	0	units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Jednostki (wartość)	1	units	Dowolna wartość tekstowa. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na „Włączone”

Parameter	type	name	value
Rozmiar	0	units_size	Możliwe wartości od 1 do 100. Domyślnie: 25. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na „Włączone”
Pogrubienie	0	units_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na „Włączone”
Pozycja	0	units_pos	0 - Przed wartością; 1 - Nad wartością; 2 - *(domyślnie)* Po wartości; 3 - Pod wartością. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na „Włączone” Ten parametr jest ignorowany, jeśli zostanie ustawiony na jedną z następujących jednostek związanych z czasem: unixtime, uptime, s.
Kolor	1	units_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Łuk wartości			
Rozmiar łuku	0	value_arc_size	Możliwe wartości od 1 do 100. Domyślnie: 20.
Wskazówka			
Kolor	1	needle_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste). Zachowanie parametru: - *obsługiwane*, jeśli ustawiono obiekt pola widżetu dashboardu *Show* z wartością „Value arc” lub *Show arc* ustawiono na „Włączone”
Skala			
Pokaż jednostki	0	scale_show_units	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Jednostki* (pole wyboru) jest ustawione na „Włączone” oraz ustawiono obiekt pola widżetu dashboardu *Show* z wartością „Value arc” lub *Show arc* ustawiono na „Włączone”
Rozmiar	0	scale_size	Możliwe wartości od 1 do 100. Domyślnie: 15. Zachowanie parametru: - *obsługiwane*, jeśli ustawiono obiekt pola widżetu dashboardu *Show* z wartością „Value arc” lub *Show arc* ustawiono na „Włączone”
Miejsca dziesiętne	0	scale_decimal_places	Możliwe wartości od 1 do 10. Domyślnie: 0. Zachowanie parametru: - *obsługiwane*, jeśli ustawiono obiekt pola widżetu dashboardu *Show* z wartością „Value arc” lub *Show arc* ustawiono na „Włączone”
Progi			
Kolor	1	thresholds.0.color	Szesnastkowy kod koloru (np. FF0000).
Próg	1	thresholds.0.threshold	Dowolna wartość liczbowa. Obsługiwane są przyrostki (np. „1d”, „2w”, „4K”, „8G”).

Parameter	type	name	value
Pokaż etykiety	0	th_show_labels	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli ustawiono *Progi* oraz ustawiono obiekt pola widżetu dashboardu *Show* z wartością „Value arc" lub *Show arc* ustawiono na „Włączone"
Pokaż łuk	0	th_show_arc	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli ustawiono *Progi*
Rozmiar łuku	0	th_arc_size	Możliwe wartości od 1 do 100. Domyślnie: 5. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż łuk* jest ustawione na „Włączone"

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Gauge*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Gauge*

Skonfiguruj widżet *Gauge*, który wyświetla wartość pozycji dla pozycji „44474" (Interface enp0s3: Bits sent). Dodatkowo dostrój wizualnie widżet za pomocą wielu zaawansowanych opcji, w tym progów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

7 Mapa geograficzna

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Geomap* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Geomap*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widżet *Geomap*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	Identyfikator grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Grupy hostów (widżet)	1	groupids._reference	Zamiast identyfikatora grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Hosty	3	hostids.0	Identyfikator hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* musi być całkowicie nieskonfigurowany albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Hosty (widżet/dashboard)	1	hostids._reference	Zamiast identyfikatora hosta: `DASHBOARD.hostids` - ustaw dashboardowy selektor hostów jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Znaczniki			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Nazwa znacznika	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Operator	0	tags.0.operator	0 - Zawiera; 1 - Jest równy; 2 - Nie zawiera; 3 - Nie jest równy; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Wartość znacznika	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Widok początkowy	1	default_view	Rozdzielane przecinkami wartości *szerokości geograficznej*, *długości geograficznej* i *poziomu powiększenia* (*opcjonalnie*, możliwe wartości od 0 do 30). Przykład: `40.6892494,-74.0466891,10`.
Grupowanie			
Tryb grupowania	0	clustering_mode	0 - *(domyślnie)* Automatyczny; 1 - Poziom powiększenia.
Poziom powiększenia grupowania	0	clustering_zoom_level	Możliwe wartości od 0 do 30. Wartość domyślna: 0. Zachowanie parametru: - *obsługiwany*, jeśli *Tryb grupowania* jest ustawiony na „Poziom powiększenia”
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikatowa w dashboardzie, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Geomap*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Geomap*

Skonfiguruj widżet *Geomap*, który wyświetla hosty z grup hostów „2” i „22” na podstawie następującej konfiguracji tagów: tag o nazwie „component” zawiera wartość „node”, a tag o nazwie „location” jest równy wartości „New York”. Dodatkowo ustaw początkowy widok mapy na współrzędne „40.6892494” (szerokość geograficzna), „-74.0466891” (długość geograficzna) z poziomem powiększenia „10”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

8 Graph

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Graph* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Graph*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Dla widżetu *Graph* obsługiwane są następujące parametry.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Zestaw danych

Poniższe parametry są obsługiwane podczas konfigurowania *Zestawu danych*.

Note:
Pierwsza liczba w nazwie właściwości (np. ds.0.hosts.0, ds.0.items.0) oznacza konkretny zestaw danych, natomiast druga liczba, jeśli występuje, oznacza skonfigurowany host lub pozycję.

Parametr	type	name	value
Typ zestawu danych	0	ds.0.dataset_type	0 - Lista pozycji; 1 - *(domyślnie)* Wzorce pozycji.
Pozycje	4	ds.0.itemids.0	Identyfikator pozycji. Podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu należy ustawić tylko pozycje skonfigurowane na szablonie. Uwaga: Aby skonfigurować wiele pozycji, utwórz obiekt pola widgetu pulpitu nawigacyjnego dla każdej pozycji, zwiększając liczbę w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na „Lista pozycji”, a *Pozycje (widget)* nie są ustawione
Pozycje (widget)	1	ds.0.itemids.0._reference	Zamiast identyfikatora pozycji: `ABCDE._itemid` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na „ABCDE”) jako źródło danych dla pozycji. Uwaga: Aby skonfigurować wiele widgetów, utwórz obiekt pola widgetu pulpitu nawigacyjnego dla każdego widgetu, zwiększając liczbę w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na „Lista pozycji”, a *Pozycje* nie są ustawione
Kolor	1	ds.0.color.0	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na „Lista pozycji”
Wzorce hostów	1	ds.0.hosts.0	Nazwa lub wzorzec hosta (np. „Zabbix*”). Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na „Wzorce pozycji” Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu.
Wzorce pozycji	1	ds.0.items.0	Nazwa lub wzorzec pozycji (np. „*: Liczba przetworzonych *wartości na sekundę”). Podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu należy ustawić tylko wzorce pozycji skonfigurowanych na szablonie. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na „Wzorce pozycji”
Kolor	1	ds.0.color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwany*, jeśli *Typ zestawu danych* jest ustawiony na „Wzorce pozycji”, a *Paleta kolorów* nie jest ustawiona

Parametr	type	name	value
Paleta kolorów	0	ds.0.color_palette	Indeks palety kolorów. Możliwe wartości mieszczą się w zakresie od 0 do 11. Wartość domyślna: 0. Zachowanie parametru: - *obsługiwany*, jeśli *Typ zestawu danych* jest ustawiony na „Wzorce pozycji”, a *Kolor* nie jest ustawiony
Rysowanie	0	ds.0.type	0 - *(domyślnie)* Linia; 1 - Punkty; 2 - Schodki; 3 - Słupki.
Skumulowane	0	ds.0.stacked	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwany*, jeśli *Rysowanie* jest ustawione na „Linia”, „Schodki” lub „Słupki”
Szerokość	0	ds.0.width	Możliwe wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 1. Zachowanie parametru: - *obsługiwany*, jeśli *Rysowanie* jest ustawione na „Linia” lub „Schodki”
Rozmiar punktu	0	ds.0.pointsize	Możliwe wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 3. Zachowanie parametru: - *obsługiwany*, jeśli *Rysowanie* jest ustawione na „Punkty”
Przezroczystość	0	ds.0.transparency	Możliwe wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 5.
Wypełnienie	0	ds.0.fill	Możliwe wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 3. Zachowanie parametru: - *obsługiwany*, jeśli *Rysowanie* jest ustawione na „Linia” lub „Schodki”
Brakujące dane	0	ds.0.missingdatafunc	0 - *(domyślnie)* Brak; 1 - Połączone; 2 - Traktuj jako 0; 3 - Ostatnia znana wartość. Zachowanie parametru: - *obsługiwany*, jeśli *Rysowanie* jest ustawione na „Linia” lub „Schodki”
Zastąp host	1	ds.0.override_hostid.ABCDE.reference	`_hostid` - ustaw kompatybilny widget (z parametrem „Reference” ustawionym na „ABCDE”) jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hostów pulpitu nawigacyjnego jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu.
Oś Y	0	ds.0.axisy	0 - *(domyślnie)* Lewa; 1 - Prawa.
Odwróć wartości	0	ds.0.invert_values	0 - (domyślnie) Wyłączone; 1 - Włączone.

Parametr	type	name	value
Przesunięcie czasu	1	ds.0.timeshift	Prawidłowy ciąg czasu (np. 3600, 1h itd.). Można używać przyrostków czasu. Dozwolone są również wartości ujemne. Wartość domyślna: "" (pusta).
Funkcja agregacji	0	ds.0.aggregate_function	0 - *(domyślnie)* nieużywana; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.
Interwał agregacji	1	ds.0.aggregate_interval	Prawidłowy ciąg czasu (np. 3600, 1h itd.). Można używać przyrostków czasu. Wartość domyślna: 1h.
Agreguj	0	ds.0.aggregate_grouping	0 - *(domyślnie)* Każda pozycja; 1 - Zestaw danych. Zachowanie parametru: - *obsługiwany*, jeśli *Funkcja agregacji* jest ustawiona na „min”, „max”, „avg”, „count”, „sum”, „first” lub „last”
Przybliżenie	0	ds.0.approximation	1 - min; 2 - *(domyślnie)* avg; 4 - max; 7 - all.
Etykieta zestawu danych	1	ds.0.data_set_label	Dowolna wartość tekstowa. Wartość domyślna: "" (pusta).
Tagi pozycji			
Typ oceny	0	ds.0.item_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	ds.0.item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów pozycji*
Operator	0	ds.0.item_tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów pozycji*
Wartość tagu	1	ds.0.item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów pozycji*

Opcje wyświetlania

Poniższe parametry są obsługiwane podczas konfigurowania *Opcji wyświetlania*.

Parametr	type	name	value
Wybór danych historycznych	0	source	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy.
Proste wyzwalacze	0	simple_triggers	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Czas pracy	0	working_time	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Nazwy hostów na etykietach	0	show_hostnames	0 - *(domyślnie)* Automatycznie; 1 - Pokaż; 2 - Ukryj. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu.
Linia percentyla (lewa)			
Stan	0	percentile_left	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zestawu danych*) jest ustawiona na „Lewa”
Wartość	0	percentile_left_value	Możliwe wartości mieszczą się w zakresie od 1 do 100. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zestawu danych*) jest ustawiona na „Lewa”
Linia percentyla (prawa)			
Stan	0	percentile_right	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zestawu danych*) jest ustawiona na „Prawa”
Wartość	0	percentile_right_value	Możliwe wartości mieszczą się w zakresie od 1 do 100. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zestawu danych*) jest ustawiona na „Prawa”

Okres czasu

Następujące parametry są obsługiwane przy konfigurowaniu *Okresu czasu*.

Parametr	type	name	value
Okres czasu	1	time_period._referenc	DASHBOARD._timeperiod - ustaw selektor okresu czasu pulpitu jako źródło danych; ABCDE._timeperiod - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych. Domyślnie: DASHBOARD._timeperiod Alternatywnie można ustawić okres czasu tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni czasu bezwzględnego (YYYY-MM-DD hh:mm:ss) lub względnego (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono time_period.to
Do	1	time_period.to	Prawidłowy ciąg czasu w składni czasu bezwzględnego (YYYY-MM-DD hh:mm:ss) lub względnego (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono time_period.from

Osie

Następujące parametry są obsługiwane przy konfigurowaniu *Osi*.

Parametr	type	name	value
Lewe Y	0	lefty	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zbiór danych*) jest ustawiona na „Lewa"
Prawe Y	0	righty	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Oś Y* (w konfiguracji *Zbiór danych*) jest ustawiona na „Prawa"
Skala	0	lefty_scale righty_scale	0 - *(domyślnie)* Liniowa; 1 - Logarytmiczna.
Min	1	lefty_min righty_min	Dowolna wartość liczbowa. Domyślnie: "" (puste).
Max	1	lefty_max righty_max	Dowolna wartość liczbowa. Domyślnie: "" (puste).
Jednostki (typ)	0	lefty_units righty_units	0 - *(domyślnie)* Auto; 1 - Statyczne.
Jednostki (wartość)	1	lefty_static_units righty_static_units	Dowolna wartość tekstowa. Domyślnie: "" (puste).

Parametr	type	name	value
Oś X	0	xaxis	0 - Wyłączona; 1 - *(domyślnie)* Włączona.

Legenda

Poniższe parametry są obsługiwane podczas konfigurowania *Legendy*.

Parametr	type	name	value
Pokaż legendę	0	legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Wyświetl min./śr./maks.	0	legend_statistic	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Pokaż funkcję agregacji	0	legend_aggregation	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Wiersze	0	legend_lines_mode	0 - *(domyślnie)* Stała; 1 - Zmienna. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Liczba wierszy/ Maksymalna liczba wierszy	0	legend_lines	Możliwe wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 1. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Liczba kolumn	0	legend_columns	Możliwe wartości mieszczą się w zakresie od 1 do 4. Wartość domyślna: 4. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”, a opcja *Wyświetl min./śr./maks.* jest ustawiona na „Wyłączone”

Problemy

Następujące parametry są obsługiwane przy konfigurowaniu *Problemów*.

Parametr	type	name	value
Pokaż problemy	0	show_problems	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Tylko wybrane pozycje	0	graph_item_problems	0 - Wyłączone; 1 - *(domyślnie)* Włączone.

Parametr	type	name	value
Hosty problemów	1	problemhosts.0	Nazwa hosta. Uwaga: Liczba w nazwie właściwości odnosi się do skonfigurowanego hosta. Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z kolejnym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Poziom ważności	0	severities.0	0 - Niesklasyfikowane; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z kolejnym numerem w nazwie właściwości.
Problem	1	problem_name	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Tagi problemu			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*

Nadpisania

Poniższe parametry są obsługiwane podczas konfigurowania *Nadpisań*.

Note:
Pierwsza liczba w nazwie właściwości (np. or.0.hosts.0, or.0.items.0) reprezentuje określony zestaw danych, natomiast druga liczba, jeśli występuje, reprezentuje skonfigurowany host lub pozycję.

Parametr	type	name	value
Wzorce hostów	1	or.0.hosts.0	Nazwa lub wzorzec hosta (np. `Zabbix*`). Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu. Zachowanie parametru: - *wymagany* podczas konfigurowania *Nadpisań*
Wzorce pozycji	1	or.0.items.0	Nazwa lub wzorzec pozycji (np. `*: Number of processed *values per second`). Podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu należy ustawić tylko wzorce pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagany* podczas konfigurowania *Nadpisań*
Kolor bazowy	1	or.0.color	Szesnastkowy kod koloru (np. `FF0000`).
Szerokość	0	or.0.width	Możliwe wartości mieszczą się w zakresie od 1 do 10.
Rysowanie	0	or.0.type	0 - Linia; 1 - Punkty; 2 - Schodkowy; 3 - Słupki.
Przezroczystość	0	or.0.transparency	Możliwe wartości mieszczą się w zakresie od 1 do 10.
Wypełnienie	0	or.0.fill	Możliwe wartości mieszczą się w zakresie od 1 do 10.
Rozmiar punktu	0	or.0.pointsize	Możliwe wartości mieszczą się w zakresie od 1 do 10.
Brakujące dane	0	or.0.missingdatafunc	0 - Brak; 1 - Połączone; 2 - Traktuj jako 0; 3 - Ostatnia znana wartość.
Oś Y	0	or.0.axisy	0 - Lewa; 1 - Prawa.
Odwróć wartości	0	or.0.invert_values	0 - (domyślnie) Wyłączone; 1 - Włączone.
Przesunięcie czasu	1	or.0.timeshift	Prawidłowy ciąg czasu (np. 3600, `1h` itd.). Można używać przyrostków czasu. Dozwolone są wartości ujemne.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Graph*. Aby uzyskać więcej informacji na temat konfiguracji pulpitu, zobacz `dashboard.create`.

Konfigurowanie widżetu *Graph*

Skonfiguruj widżet *Graph* w następujący sposób:

- 2 zestawy danych dla łącznie 9 pozycji na 1 hoście.
- Pierwszy zestaw danych jest typu „Item list" i składa się z 3 pozycji, które są reprezentowane przez linie o różnych kolorach, ale tej samej szerokości, przezroczystości i wypełnieniu.
- Drugi zestaw danych jest typu „Item patterns", składa się z 6 pozycji, ma skonfigurowaną agregację i jest reprezentowany przez linię z niestandardowym kolorem, szerokością, przezroczystością i wypełnieniem.
- Drugi zestaw danych ma również niestandardową etykietę zestawu danych.
- Dane na wykresie są wyświetlane dla okresu ostatnich 3 godzin.
- Problemy na wykresie są wyświetlane tylko dla skonfigurowanych pozycji.
- Wykres ma dwie osie Y, z których prawa oś Y wyświetla wartości tylko dla drugiego zestawu danych.
- Legenda wykresu wyświetla skonfigurowane pozycje w 4 wierszach, a także minimalne, maksymalne i średnie wartości zestawów danych.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

9 Graph (classic)

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Graph (classic)* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Graph (classic)*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Wykres (klasyczny)*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Źródło	0	source_type	0 - *(domyślnie)* Wykres; 1 - Wykres prosty.
Wykres	6	graphid.0	ID wykresu. Zachowanie parametru: - *wymagany*, jeśli *Źródło* jest ustawione na "Wykres"
Wykres (widżet)	1	graphid._reference	Zamiast ID wykresu: `ABCDE._graphid` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla wykresów. Zachowanie parametru: - *wymagany*, jeśli *Źródło* jest ustawione na "Wykres prosty", a *Wykres* nie jest ustawiony
Pozycja	4	itemid.0	ID pozycji. Zachowanie parametru: - *wymagany*, jeśli *Źródło* jest ustawione na "Wykres prosty", a *Pozycja (widżet)* nie jest ustawiona
Pozycja (widżet)	1	itemid._reference	Zamiast ID pozycji: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla pozycji. Zachowanie parametru: - *wymagany*, jeśli *Źródło* jest ustawione na "Wykres prosty", a *Pozycja* nie jest ustawiona

Parametr	type	name	value
Okres czasu	1	time_period._referenceDASHBOARD._timeperiod	DASHBOARD._timeperiod - ustaw selektor okresu czasu pulpitu jako źródło danych; ABCDE._timeperiod - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych. Domyślnie: DASHBOARD._timeperiod Alternatywnie możesz ustawić okres czasu tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub względnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono time_period.to
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub względnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono time_period.from
Pokaż legendę	0	show_legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Nadpisz host	1	override_hostid._referenceABCDE._hostid	ABCDE._hostid - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla hostów; DASHBOARD._hostid - ustaw selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. ABCDE lub JBPNL). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Wykres (klasyczny)*. Aby uzyskać więcej informacji na temat konfigurowania pulpitu nawigacyjnego, zobacz dashboard.create.

Konfigurowanie widżetu *Graph (classic)*

Skonfiguruj widżet *Graph (classic)*, który wyświetla prosty wykres dla pozycji „42269".

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

10 Graph prototype

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Graph prototype* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Graph prototype*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widżet *Prototyp wykresu*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Źródło	0	source_type	2 - *(domyślnie)* Prototyp wykresu; 3 - Prototyp prostego wykresu.
Prototyp wykresu	7	graphid.0	Identyfikator protokołu wykresu. Zachowanie parametru: - *wymagany*, jeśli *Źródło* jest ustawione na „Prototyp wykresu”
Prototyp pozycji	5	itemid.0	Identyfikator protokołu pozycji. Zachowanie parametru: - *wymagany*, jeśli *Źródło* jest ustawione na „Prototyp prostego wykresu”
Okres	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor okresu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na „`ABCDE`”) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli parametr *Okres* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.to`
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli parametr *Okres* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.from`
Pokaż legendę	0	show_legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Zastąp host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na „`ABCDE`”) jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hostów dashboardu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Kolumny	0	columns	Możliwe wartości mieszczą się w zakresie od 1 do 24. Domyślnie: 2.
Wiersze	0	rows	Możliwe wartości mieszczą się w zakresie od 1 do 16. Domyślnie: 1.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikatowa w dashboardzie, do którego należy widżet. Zachowanie parametru: - *wymagane*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Prototyp wykresu*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Graph prototype*

Skonfiguruj widżet *Graph prototype*, który wyświetla siatkę 3 wykresów (3 kolumny, 1 wiersz) utworzonych z prototypu pozycji (ID: ”42316”) przez wykrywanie niskopoziomowe.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "graphprototype",
                        "name": "Graph prototype",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

11 Honeycomb

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Honeycomb* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Honeycomb*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Honeycomb*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (Widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw zgodny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z numerem zwiększanym w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty (Widżet/Pulpit)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor hostów pulpitu jako źródło danych dla hostów; `ABCDE._hostids` - ustaw zgodny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Tagi hostów			
Typ oceny	0	evaltype_host	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Nazwa tagu	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wartość tagu	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wzorce pozycji	1	items.0	Nazwa lub wzorzec pozycji. Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu pulpitu dla każdego wzorca pozycji z numerem zwiększanym w nazwie właściwości. Zachowanie parametru: - *wymagane*
Tagi pozycji			
Typ oceny	0	evaltype_item	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów pozycji*
Operator	0	item_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów pozycji*

Parametr	type	name	value
Wartość tagu	1	item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów pozycji*
Pokaż hosty w konserwacji	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Pokaż	0	show.0	1 - Etykieta podstawowa; 2 - Etykieta dodatkowa. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z numerem zwiększanym w nazwie właściwości. Domyślnie: 1, 2.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (na przykład `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Zaawansowana konfiguracja

Następujące zaawansowane parametry konfiguracji są obsługiwane przez widżet *Honeycomb*.

Note:
Liczba w nazwie właściwości *Thresholds* (np. thresholds.0.color) odwołuje się do miejsca progu na liście posortowanej rosnąco. Jeśli jednak progi zostaną skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontend Zabbix (np. `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parameter	type	name	value
Etykieta główna			
Typ	0	primary_label_type	0 - *(domyślnie)* Tekst; 1 - Wartość.
Tekst	1	primary_label	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, makra użytkownika. Domyślnie: `{HOST.NAME}` Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Tekst”
Miejsca dziesiętne	0	primary_label_decimal_places	Możliwe wartości mieszczą się w zakresie od 0 do 6. Domyślnie: 2. Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”
Rozmiar (typ)	0	primary_label_size_type	0 - *(domyślnie)* Automatyczny; 1 - Niestandardowy.

Parameter	type	name	value
Rozmiar	0	primary_label_size	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 20. Zachowanie parametru: - *obsługiwany*, jeśli *Rozmiar* (typ) jest ustawiony na „Niestandardowy”
Pogrubienie	0	primary_label_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	primary_label_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: na podstawie wartości `theme` obiektu Settings i obiektu User: `1F2C33` dla „blue-theme”, „blue-classic-theme” lub „hc-light”; `EEEEEE` dla „dark-theme”, „dark-classic-theme” lub „hc-dark”; `E5F1FF` dla „dark-blue-theme”.
Jednostki (pole wyboru)	0	primary_label_units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”
Jednostki (wartość)	1	primary_label_units	Dowolna wartość tekstowa. "" (pusta) Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”, a *Jednostki* (pole wyboru) są ustawione na „Włączone”
Pozycja	0	primary_label_units_pos	0 - Przed wartością; 1 - *(domyślnie)* Po wartości. Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”, a *Jednostki* (pole wyboru) są ustawione na „Włączone” Ten parametr jest ignorowany, jeśli zostanie ustawiony na jedną z następujących jednostek związanych z czasem: unixtime, uptime, s.
Etykieta dodatkowa			
Typ	0	secondary_label_type	0 - Tekst; 1 - *(domyślnie)* Wartość.
Tekst	1	secondary_label	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, makra użytkownika. Domyślnie: `{{ITEM.LASTVALUE}.fmtnum(2)}` Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Tekst”
Miejsca dziesiętne	0	secondary_label_decimal_places	Możliwe wartości mieszczą się w zakresie od 0 do 6. Domyślnie: 2. Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”
Rozmiar (typ)	0	secondary_label_size_type	0 - *(domyślnie)* Automatyczny; 1 - Niestandardowy.
Rozmiar	0	secondary_label_size	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 30. Zachowanie parametru: - *obsługiwany*, jeśli *Rozmiar* (typ) jest ustawiony na „Niestandardowy”

Parameter	type	name	value
Pogrubienie	0	secondary_label_bold	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Kolor	1	secondary_label_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: na podstawie wartości `theme` obiektu Settings i obiektu User: `1F2C33` dla „blue-theme”, „blue-classic-theme” lub „hc-light”; `EEEEEE` dla „dark-theme”, „dark-classic-theme” lub „hc-dark”; `E5F1FF` dla „dark-blue-theme”.
Jednostki (pole wyboru)	0	secondary_label_units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”
Jednostki (wartość)	1	secondary_label_units	Dowolna wartość tekstowa. "" (pusta) Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”, a *Jednostki* (pole wyboru) są ustawione na „Włączone”
Pozycja	0	secondary_label_position	0 - Przed wartością; 1 - *(domyślnie)* Po wartości. Zachowanie parametru: - *obsługiwany*, jeśli *Typ* jest ustawiony na „Wartość”, a *Jednostki* (pole wyboru) są ustawione na „Włączone” Ten parametr jest ignorowany, jeśli zostanie ustawiony na jedną z następujących jednostek związanych z czasem: unixtime, uptime, s.
Kolor tła			
Kolor tła	1	bg_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: na podstawie wartości `theme` obiektu Settings i obiektu User: `D9E7ED` dla „blue-theme” i „blue-classic-theme”; `3D5059` dla „dark-theme” i „dark-classic-theme”; `004FA8` dla „dark-blue-theme”; `AAD7E9` dla „hc-light”; `335463` dla „hc-dark”.
Progi			
Interpolacja kolorów	0	interpolation	0 - Wyłączona; 1 - *(domyślnie)* Włączona.
Kolor	1	thresholds.0.color	Szesnastkowy kod koloru (np. `FF0000`).
Próg	1	thresholds.0.threshold	Dowolna wartość liczbowa. Obsługiwane są przyrostki (np. „1d”, „2w”, „4K”, „8G”).

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Honeycomb*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Honeycomb*

Skonfiguruj widżet *Honeycomb*, który wyświetla wykorzystanie procesów serwera Zabbix. Dodatkowo zmień główną etykietę komórek honeycomb i dostrój wizualnie widżet za pomocą progów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
```

```
        "name": "My dashboard",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

12 Dostępność hosta

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widżetu dashboardu umożliwiają skonfigurowanie widżetu *Dostępność hosta* w metodach `dashboard.create` i `dashboard.update`.

Parametry

Następujące parametry są obsługiwane dla widżetu *Dostępność hosta*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Typ interfejsu	0	interface_type.0	0 - Brak; 1 - agent Zabbix (sprawdzanie pasywne); 2 - SNMP; 3 - IPMI; 4 - JMX; 5 - agent Zabbix (sprawdzanie aktywne). Domyślnie: 1, 2, 3, 4, 5 (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z numerem zwiększanym w nazwie właściwości.
Układ	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.
Pokaż hosty w konserwacji	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Pokaż tylko sumy	0	only_totals	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Dostępność hosta*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Dostępność hosta*

Skonfiguruj widżet *Dostępność hosta*, który wyświetla informacje o dostępności (w układzie pionowym) dla hostów z grupy hostów „4" ze skonfigurowanymi interfejsami „Zabbix agent" i „SNMP".

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
```

```
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

######13 Karta hosta {#manual-api-reference-dashboard-widget_fields-host_card}

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Karta hosta* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Karta hosta*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Karta hosta*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Host	3	hostid.0	ID hosta. Zachowanie parametrów: - *wymagane*, jeśli *Host (Widżet/Pulpit)* nie jest ustawiony Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Host (Widżet/Pulpit)	1	hostid._reference	Zamiast ID hosta: `DASHBOARD.hostid` - ustaw selektor hosta pulpitu jako źródło danych dla hosta; `ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hosta. Zachowanie parametrów: - *wymagane*, jeśli *Host* nie jest ustawiony Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Pokaż ukryte problemy	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Parametr	type	name	value
Pokaż	0	sections.0	0 - Grupy hostów; 1 - Opis; 2 - Monitorowanie; 3 - Dostępność; 4 - Monitorowane przez; 5 - Szablony; 6 - Inwentarz; 7 - Tagi. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności sekcji na liście sekcji. Aby skonfigurować wiele sekcji, utwórz obiekt pola widżetu pulpitu dla każdej sekcji, zwiększając numer w nazwie właściwości.

Następujące parametry są obsługiwane, jeśli *Pokaż* jest ustawione na „Inwentarz”.

Parametr	type	name	value
Pola inwentarza	0	inventory.0	ID inwentarza. Uwaga: Aby skonfigurować wiele pól inwentarza, utwórz obiekt pola widżetu pulpitu dla każdego pola inwentarza, zwiększając numer w nazwie właściwości.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Host card*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Host card*

Skonfiguruj widżet *Host card*, który wyświetla następujące sekcje: „Monitoring”, „Availability”, „Monitored by”, „Inventory” i „Tags”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostcard",
                        "name": "Host card",
                        "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
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
```

```
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

14 Nawigator hostów

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Host navigator* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Host navigator*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widżet *Nawigator hostów*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	Identyfikator grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Grupy hostów (widżet)	1	groupids._reference	Zamiast identyfikatora grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Wzorce hostów	1	hosts.0	Nazwa lub wzorzec hosta. Uwaga: Aby skonfigurować wiele wzorców hostów, utwórz obiekt pola widżetu dashboardu dla każdego wzorca hosta, zwiększając numer w nazwie właściwości. W przypadku wielu wzorców hostów parametr *Grupy hostów* musi być całkowicie nieskonfigurowany albo skonfigurowany z co najmniej jedną grupą hostów, do której należą hosty pasujące do skonfigurowanych wzorców hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Stan hosta	0	status	-1 - *(domyślnie)* Dowolny; 0 - Włączony; 1 - Wyłączony. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Tagi hosta			
Typ oceny	0	host_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Nazwa tagu	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Jest równy; 2 - Nie zawiera; 3 - Nie jest równy; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Wartość tagu	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Ważność	0	severities.0	0 - Niesklasyfikowane; 1 - Informacyjne; 2 - Ostrzeżenie; 3 - Średnie; 4 - Wysokie; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Pokaż hosty w konserwacji	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Pokaż problemy	0	show_problems	0 - Wszystkie; 1 - *(domyślnie)* Niewstrzymane; 2 - Brak.
Grupuj według			
Atrybut	0	group_by.0.attribute	0 - Grupa hostów; 1 - Wartość tagu; 2 - Ważność. Uwaga: Numer w nazwie właściwości wskazuje kolejność atrybutu na liście atrybutów grupowania. Zachowanie parametru: - *wymagany* podczas konfigurowania *Grupuj według*
Wartość	1	group_by.0.tag_name	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje zestaw atrybutu grupowania określony w parametrze *Atrybut*. Zachowanie parametru: - *wymagany* podczas konfigurowania *Grupuj według*, gdy parametr *Atrybut* ma wartość „Wartość tagu”
Limit hostów	0	show_lines	Możliwe wartości mieszczą się w zakresie od 1 do 9999. Domyślnie: 100. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Wartość ta musi być unikatowa w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Host navigator*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Host navigator*

Skonfiguruj widżet *Host navigator*, który wyświetla hosty pogrupowane według ich grupy hostów, a następnie według wartości tagu „city”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": "30",
        "auto_start": "1",
        "pages": [
            {
                "widgets": [
                    {
                        "type": "hostnavigator",
                        "name": "Host navigator",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

15 Karta pozycji

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Karta pozycji* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Karta pozycji*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Karta pozycji*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Pozycja	4	itemid.0	ID pozycji. Zachowanie parametrów: - *wymagane*, jeśli *Pozycja (widżet)* nie jest ustawiona
Pozycja (widżet)	1	itemid._reference	Zamiast ID pozycji: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla pozycji. Zachowanie parametrów: - *wymagane*, jeśli *Pozycja* nie jest ustawiona

Parametr	type	name	value
Pokaż	0	sections.0	0 - Opis; 1 - Tekst błędu; 2 - Metryki; 3 - Najnowsze dane; 4 - Typ informacji; 5 - Wyzwalacze; 6 - Interfejs hosta; 7 - Typ; 8 - Inwentarz hosta; 9 - Tagi. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności sekcji na liście sekcji. Aby skonfigurować wiele sekcji, utwórz obiekt pola widżetu pulpitu dla każdej sekcji z kolejnym numerem zwiększonym w nazwie właściwości.
Nadpisz host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hosta pulpitu jako źródło danych dla hostów. Domyślnie: "" (puste) Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Sparkline

Poniższe parametry są obsługiwane, jeśli opcja *Show* jest ustawiona na „Latest data".

Parameter	type	name	value
Width	0	sparkline.width	Dozwolone wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 1.
Fill	0	sparkline.fill	Dozwolone wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 3.
Color	1	sparkline.color	Szesnastkowy kod koloru (np. `FF0000`). Wartość domyślna: `42A5F5`.
Time period	1	sparkline.time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor okresu czasu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem `reference` równym `ABCDE`) jako źródło danych. Wartość domyślna: "" (pusta) Alternatywnie możesz ustawić okres czasu tylko w parametrach *From* i *To*.
From	1	sparkline.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Wartość domyślna: now-1h. Zachowanie parametru: - *obsługiwany*, jeśli parametr *Time period* nie jest ustawiony - *wymagany*, jeśli ustawiono `sparkline.time_period.to`

Parameter	type	name	value
To	1	sparkline.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Wartość domyślna: now. Zachowanie parametru: - *obsługiwany*, jeśli parametr *Time period* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.to`
History data	0	sparkline.history	0 - *(domyślnie)* Auto; 1 - History; 2 - Trends.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Karta pozycji*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Karta pozycji*

Skonfiguruj widżet *Karta pozycji*, który wyświetla następujące sekcje: „Opis”, „Najnowsze dane”, „Wyzwalacze” i „Tagi”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "itemcard",
                        "name": "Item card",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

16 Historia pozycji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Historia pozycji* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Historia pozycji*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widget *Historia pozycji*.

Parametr	typ	nazwa	wartość
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Układ	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.
Kolumny (zobacz poniżej)			
Pokaż wiersze	0	show_lines	Dozwolone wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 25.
Zastąp host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hostów dashboardu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Konfiguracja zaawansowana (zobacz poniżej)			
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikatowa w dashboardzie, do którego należy widget. Zachowanie parametru: - *wymagany*

Kolumny

Kolumny mają wspólne parametry oraz dodatkowe parametry zależne od konfiguracji parametru *Pozycja*.

Note:
Dla wszystkich parametrów związanych z kolumnami liczba w nazwie właściwości (np. columns.0.name) odnosi się do kolumny, dla której parametr jest skonfigurowany.

Następujące parametry są obsługiwane dla wszystkich kolumn.

Parametr	type	name	value
Nazwa	1	columns.0.name	Dowolna wartość tekstowa. Zachowanie parametru: - *wymagane*
Pozycja	4	columns.0.itemid	ID pozycji. Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko pozycje skonfigurowane w szablonie. Zachowanie parametru: - *wymagane*
Kolor bazowy	1	columns.0.base_color	Szesnastkowy kod koloru (np. `FF0000`). Domyślnie: `""` (puste).

Następujące parametry kolumn są obsługiwane, jeśli skonfigurowana *pozycja* jest pozycją typu numerycznego.

Parametr	type	name	value
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Bez zmian; 2 - Pasek; 3 - Wskaźniki.
Min	1	columns.0.min	Dowolna wartość liczbowa. Zachowanie parametru: - *obsługiwane*, jeśli *Wyświetlanie* jest ustawione na „Pasek” lub „Wskaźniki”
Max	1	columns.0.max	Dowolna wartość liczbowa. Zachowanie parametru: - *obsługiwane*, jeśli *Wyświetlanie* jest ustawione na „Pasek” lub „Wskaźniki”
Progi			
Kolor	1	columns.0.thresholds.color	Szesnastkowy kod koloru (np. FF0000).
Próg	1	columns.0.thresholds.threshold	Dowolna wartość liczbowa. Sufiksy (np. „1d”, „2w”, „4K”, „8G”) są obsługiwane.
Dane historyczne	0	columns.0.history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.

Następujące parametry kolumn są obsługiwane, jeśli skonfigurowana *pozycja* jest pozycją typu znakowego, tekstowego lub dziennika.

Parameter	type	name	value
Wyróżnienia			
Wyróżnienie	1	columns.0.highlights.color	Szesnastkowy kod koloru (np. FF0000).
Próg	1	columns.0.highlights.pattern	Dowolne wyrażenie regularne.
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Tak jak jest; 4 - HTML; 5 - Jeden wiersz.
Jeden wiersz	0	columns.0.max_length	Możliwe wartości mieszczą się w zakresie od 1 do 500. Domyślnie: 100. Zachowanie parametru: - *obsługiwany*, jeśli dla opcji *Wyświetlanie* ustawiono wartość „Jeden wiersz”
Użyj czcionki o stałej szerokości znaków	0	columns.0.monospace_font	0 - *(domyślnie)* Użyj domyślnej czcionki; 1 - Użyj czcionki o stałej szerokości znaków.
Wyświetlaj czas lokalny	0	columns.0.local_time	0 - *(domyślnie)* Wyświetlaj znacznik czasu; 1 - Wyświetlaj czas lokalny. Zachowanie parametru: - *obsługiwany*, jeśli *pozycja* jest ustawiona jako pozycja typu dziennika, a dla opcji *Pokaż znacznik czasu* ustawiono wartość „Włączone”

Następujące parametry kolumn są obsługiwane, jeśli skonfigurowana *pozycja* jest pozycją typu binarnego.

Parametr	type	nazwa	wartość
Pokaż miniaturę	1	columns.0.show_thumbnail	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Konfiguracja zaawansowana

Poniższe zaawansowane parametry konfiguracji są obsługiwane przez widżet *Historia pozycji*.

Parametr	type	name	value
Nowe wartości	0	sortorder	0 - *(domyślnie)* U góry; 1 - Na dole.
Pokaż znacznik czasu	0	show_timestamp	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Pokaż nagłówek kolumny	0	show_column_header	0 - Wył.; 1 - Poziomo; 2 - *(domyślnie)* Pionowo.
Okres czasu	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor okresu czasu pulpitu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.to`
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.from`

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Item history*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Historia pozycji*

Skonfiguruj widżet *Historia pozycji*, który wyświetla najnowsze dane dla dwóch numerycznych pozycji „42269" i „42270". Dodatkowo skonfiguruj kolumny pozycji tak, aby były wyświetlane pionowo, z nazwami kolumn wyświetlanymi poziomo; ogranicz widok do 15 wierszy danych i uwzględnij osobną kolumnę znacznika czasu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
```

```
                    "type": "itemhistory",
                    "name": "Item history",
                    "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": "CPU utilization"
                        },
                        {
                            "type": "4",
                            "name": "columns.0.itemid",
                            "value": "42269"
                        },
                        {
                            "type": "1",
                            "name": "columns.1.name",
                            "value": "Memory utilization"
                        },
                        {
                            "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

17 Nawigator pozycji

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Item navigator* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Item navigator*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widżet *Nawigator pozycji*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	Identyfikator grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Grupy hostów (widżet)	1	groupids._reference	Zamiast identyfikatora grupy hostów: `ABCDE._hostgroupids` - ustaw zgodny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Hosty	3	hostids.0	Identyfikator hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* musi być całkowicie nieskonfigurowany albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Hosty (widżet/dashboard)	1	hostids._reference	Zamiast identyfikatora hosta: `DASHBOARD.hostid` - ustaw selektor hosta dashboardu jako źródło danych dla hostów; `ABCDE._hostid` - ustaw zgodny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Tagi hostów			
Typ oceny	0	host_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Nazwa tagu	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Parametr	type	name	value
Wartość tagu	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.
Wzorce pozycji	1	items.0	Nazwa lub wzorzec pozycji. Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu dashboardu dla każdego wzorca pozycji, zwiększając numer w nazwie właściwości.
Tagi pozycji			
Typ oceny	0	item_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów pozycji*
Operator	0	item_tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów pozycji*
Wartość tagu	1	item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów pozycji*
Stan	0	state	-1 - *(domyślnie)* Wszystkie; 0 - Normalny; 1 - Nieobsługiwane.
Pokaż problemy	0	show_problems	0 - Wszystkie; 1 - *(domyślnie)* Niewyciszone; 2 - Brak.
Grupuj według			

Parametr	type	name	value
Atrybut	0	group_by.0.attribute	0 - Grupa hostów; 1 - Nazwa hosta; 2 - Wartość tagu hosta; 3 - Wartość tagu pozycji. Uwaga: Numer w nazwie właściwości wskazuje kolejność atrybutu na liście atrybutów grupowania. Zachowanie parametru: - *wymagany* podczas konfigurowania *Grupuj według*
Wartość	1	group_by.0.tag_name	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje zestaw atrybutów grupowania określony w parametrze *Atrybut*. Zachowanie parametru: - *wymagany* podczas konfigurowania *Grupuj według*, gdy parametr *Atrybut* jest ustawiony na „Wartość tagu hosta” lub „Wartość tagu pozycji”
Limit pozycji	0	show_lines	Dozwolone wartości mieszczą się w zakresie od 1 do 9999. Wartość domyślna: 100.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikatowa w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Item navigator*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Nawigator pozycji*

Skonfiguruj widżet *Nawigator pozycji*, który wyświetla do 1000 pozycji pogrupowanych według ich hosta, a następnie według wartości tagu pozycji „component”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- dashboard.create
- dashboard.update

18 Wartość pozycji

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu umożliwiają skonfigurowanie widgetu *Wartość pozycji* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Wartość pozycji*, zapoznaj się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Wartość pozycji*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Pozycja	4	itemid.0	ID pozycji. Zachowanie parametrów: - *wymagane*, jeśli *Pozycja (widżet)* nie jest ustawiona
Pozycja (widżet)	1	itemid._reference	Zamiast ID pozycji: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla pozycji. Zachowanie parametrów: - *wymagane*, jeśli *Pozycja* nie jest ustawiona
Pokaż	0	show.0	1 - Opis; 2 - Wartość; 3 - Czas; 4 - Wskaźnik zmiany; 5 - Wykres przebiegu. Domyślnie: 1, 2, 3, 4. Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z inkrementowanym numerem w nazwie właściwości.
Nadpisz host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Zaawansowana konfiguracja

Dla widżetu *Wartość pozycji* obsługiwane są następujące zaawansowane parametry konfiguracji.

Note:
Liczba w nazwie właściwości *Thresholds* (na przykład thresholds.0.color) odnosi się do miejsca progu na liście posortowanej rosnąco. Jeśli jednak progi zostaną skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontend Zabbix (na przykład `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parameter	type	name	value
Kolor tła	1	bg_color	Szesnastkowy kod koloru (na przykład `FF0000`). Domyślnie: `""` (puste).
Progi			
Kolor	1	thresholds.0.color	Szesnastkowy kod koloru (na przykład `FF0000`).
Próg	1	thresholds.0.threshold	Dowolna wartość tekstowa.
Funkcja agregacji	0	aggregate_function	0 - *(domyślnie)* nieużywana; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.
Okres czasu	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor okresu czasu pulpitu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *Od* i *Do*. Zachowanie parametru: - *obsługiwany*, jeśli *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last”
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony, a *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *wymagany*, jeśli ustawiono `time_period.to`
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony, a *Funkcja agregacji* jest ustawiona na ”min”, ”max”, ”avg”, ”count”, ”sum”, ”first”, ”last” - *wymagany*, jeśli ustawiono `time_period.from`
Dane historyczne	0	history	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.

Opis

Poniższe zaawansowane parametry konfiguracji są obsługiwane, jeśli opcja *Pokaż* jest ustawiona na „Opis”.

Parametr	type	name	value
Opis	1	description	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: `{HOST.*}`, `{ITEM.*}`, `{INVENTORY.*}`, makra użytkownika. Wartość domyślna: `{ITEM.NAME}`.

Parametr	type	name	value
Pozycja pozioma	0	desc_h_pos	0 - Lewo; 1 - *(domyślnie)* Środek; 2 - Prawo. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Pozycja pionowa	0	desc_v_pos	0 - Góra; 1 - Środek; 2 - *(domyślnie)* Dół. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Rozmiar	0	desc_size	Dozwolone wartości mieszczą się w zakresie od 1 do 100. Wartość domyślna: 15.
Pogrubienie	0	desc_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	desc_color	Szesnastkowy kod koloru (np. FF0000). Wartość domyślna: "" (pusta).

Wartość

Poniższe zaawansowane parametry konfiguracji są obsługiwane, jeśli opcja *Pokaż* jest ustawiona na „Wartość”.

Parametr	type	name	value
Miejsca dziesiętne			
Miejsca dziesiętne	0	decimal_places	Możliwe wartości mieszczą się w zakresie od 1 do 10. Domyślnie: 2.
Rozmiar	0	decimal_size	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 35.
Pozycja			
Pozycja pozioma	0	value_h_pos	0 - Lewa; 1 - *(domyślnie)* Środek; 2 - Prawa. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Pozycja pionowa	0	value_v_pos	0 - Góra; 1 - *(domyślnie)* Środek; 2 - Dół. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Rozmiar	0	value_size	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 45.
Pogrubienie	0	value_bold	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Kolor	1	value_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (pusty).
Jednostki			
Jednostki (pole wyboru)	0	units_show	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Jednostki (wartość)	1	units	Dowolna wartość tekstowa.

Parametr	type	name	value
Pozycja	0	units_pos	0 - Przed wartością; 1 - Nad wartością; 2 - *(domyślnie)* Po wartości; 3 - Pod wartością.
Rozmiar	0	units_size	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 35.
Pogrubienie	0	units_bold	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Kolor	1	units_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (pusty).

Czas

Poniższe zaawansowane parametry konfiguracji są obsługiwane, jeśli opcja *Pokaż* jest ustawiona na „Czas”.

Parametr	type	name	value
Pozycja pozioma	0	time_h_pos	0 - Lewo; 1 - *(domyślnie)* Środek; 2 - Prawo. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Pozycja pionowa	0	time_v_pos	0 - *(domyślnie)* Góra; 1 - Środek; 2 - Dół. Dwa lub więcej elementów (Opis, Wartość, Czas) nie może mieć tej samej *Pozycji poziomej* i *Pozycji pionowej*.
Rozmiar	0	time_size	Dozwolone wartości mieszczą się w zakresie od 1 do 100. Wartość domyślna: 15.
Pogrubienie	0	time_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolor	1	time_color	Szesnastkowy kod koloru (np. FF0000). Wartość domyślna: "" (pusta).

Wskaźnik zmiany

Następujące zaawansowane parametry konfiguracji są obsługiwane, jeśli dla opcji *Pokaż* ustawiono wartość „Wskaźnik zmiany”.

Parametr	type	name	value
Kolor wskaźnika zmiany ↑	1	up_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Kolor wskaźnika zmiany ↓	1	down_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).
Kolor wskaźnika zmiany ↕	1	updown_color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: "" (puste).

Sparkline

Poniższe zaawansowane parametry konfiguracji są obsługiwane, jeśli dla opcji *Show* ustawiono wartość „Sparkline”.

Parameter	type	name	value
Width	0	sparkline.width	Dozwolone wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 1.
Fill	0	sparkline.fill	Dozwolone wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 3.
Color	1	sparkline.color	Szesnastkowy kod koloru (np. `FF0000`). Wartość domyślna: `42A5F5`.
Time period	1	sparkline.time_period.reference	`DASHBOARD._timeperiod` - ustaw selektor okresu czasu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem `reference` równym `ABCDE`) jako źródło danych. Wartość domyślna: `DASHBOARD._timeperiod` Alternatywnie możesz ustawić okres czasu wyłącznie w parametrach *From* i *To*.
From	1	sparkline.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *wymagany*, jeśli ustawiono `sparkline.time_period.to`
To	1	sparkline.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *wymagany*, jeśli ustawiono `sparkline.time_period.from`
History data	0	sparkline.history	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Wartość pozycji*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Wartość pozycji*

Skonfiguruj widżet *Wartość pozycji*, który wyświetla wartość pozycji dla pozycji „42266” (dostępność agenta Zabbix). Dodatkowo dostosuj wizualnie widżet za pomocą wielu zaawansowanych opcji, w tym dynamicznego koloru tła, który zmienia się w zależności od stanu dostępności agenta Zabbix.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "item",
                        "name": "Item value",
                        "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": "Agent status"
                    },
                    {
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

19 Mapa

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Map* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Map*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Mapa*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Mapa	8	sysmapid.0	ID mapy. Zachowanie parametru: - *wymagany*, jeśli *Mapa (widżet)* nie jest ustawiona
Mapa (widżet)	1	sysmapid._reference	`ABCDE._mapid` - ustawia widżet *Drzewo nawigacji map* (z parametrem *Odwołanie* ustawionym na "ABCDE") jako źródło danych dla map. Zachowanie parametru: - *wymagany*, jeśli *Mapa* nie jest ustawiona

Parametr	type	name	value
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Mapa*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Mapa*

Skonfiguruj widżet *Mapa*, który wyświetla mapę „1”.

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "dashboard.create",
"parametry": {
"nazwa": "Mój pulpit",
"okres_wyświetlania": 30,
"automatyczny_start": 1,
"strony": [
{
"widżety": [
{
"typ": "mapa",
"nazwa": "Mapa",
"x": 0,
"y": 0,
"szerokość": 54,
"wysokość": 5,
"tryb_widoku": 0,
"pola": [
{
"typ": 8,
"nazwa": "sysmapid.0",
"wartość": 1
}
]
}
]
}
],
"grupy_użytkowników": [
{
"usrgrpid": 7,
"permission": 2
}
],
"users": [
{
"userid": 1,
"permission": 3
}
]
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"3"
]
},
"id": 1
}
```

Konfigurowanie połączonego widżetu *Mapa*

Skonfiguruj widżet *Mapa* połączony z widżetem *Drzewo nawigacji mapy*.

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "dashboard.create",
"parametry": {
"nazwa": "Mój pulpit",
"okres_wyświetlania": 30,
"automatyczny_start": 1,
"strony": [
{
"widżety": [
{
"typ": "mapa",
"nazwa": "Mapa",
"x": 0,
"y": 5,
"szerokość": 54,
"wysokość": 5,
"tryb_widoku": 0,
"pola": [
{
"typ": 1,
"nazwa": "sysmapid._reference",
"wartość": "ABCDE._mapid"
}
]
},
{
"typ": „navtree”,
„name”: „Drzewo nawigacji mapy”,
„x”: 0,
„y”: 0,
„width”: 18,
„height”: 5,
„view_mode”: 0,
„fields”: [
{
„type”: 1,
„name”: „navtree.1.name”,
„value”: „Element A”
},
{
„type”: 1,
„name”: „navtree.2.name”,
„value”: „Element B”
},
{
„type”: 1,
„name”: „navtree.3.name”,
```

```
„value”: „Element C”
},
{
„type”: 1,
„name”: „navtree.4.name”,
„value”: „Element A1”
},
{
„type”: 1,
„name”: „navtree.5.name”,
„wartość”: „Element A2”
},
{
„typ”: 1,
„nazwa”: „navtree.6.name”,
„wartość”: „Element B1”
},
{
„typ”: 1,
„nazwa”: „navtree.7.name”,
„wartość”: „Element B2”
},
{
„typ”: 0,
„nazwa”: „navtree.4.parent”,
„wartość”: 1
},
{
„typ”: 0,
„nazwa”: „navtree.5.parent”,
„wartość”: 1
},
{
„typ”: 0,
„nazwa”: „navtree.6.parent”,
„wartość”: 2
},
{
„typ”: 0,
„nazwa”: „navtree.7.parent”,
„wartość”: 2
},
{
"typ": 0,
"nazwa": "navtree.1.order",
"wartość": 1
},
{
"typ": 0,
"nazwa": "navtree.2.order",
"wartość": 2
},
{
"typ": 0,
"nazwa": "navtree.3.order",
"wartość": 3
},
{
"typ": 0,
"nazwa": "navtree.4.order",
"wartość": 1
},
```

```
{
"typ": 0,
"nazwa": "navtree.5.order",
"wartość": 2
},
{
"typ": 0,
"nazwa": "navtree.6.order",
"wartość": 1
},
{
"typ": 0,
"nazwa": "navtree.7.order",
"wartość": 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
}
```


Odpowiedź:


```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"3"
]
},
"id": 1
}
```


Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`
- Drzewo nawigacyjne mapy

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu umożliwiają skonfigurowanie widgetu *Drzewo nawigacji mapy* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Drzewo nawigacji mapy*, zapoznaj się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Drzewo nawigacji map*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Pokaż niedostępne mapy	1	show_unavailable	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametrów: - *wymagane*

Następujące parametry są obsługiwane przy konfigurowaniu elementów drzewa nawigacji map.

Parametr	type	name	value
Nazwa	1	navtree.1.name	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości określa numer elementu.
Powiązana mapa	8	navtree.1.sysmapid	ID mapy. Uwaga: Liczba w nazwie właściwości odnosi się do elementu, z którym mapa jest powiązana.
Parametry tworzenia hierarchii elementów	0	navtree.1.parent	Numer elementu nadrzędnego. Uwaga: Liczba w nazwie właściwości odnosi się do elementu podrzędnego. Wartość właściwości odnosi się do elementu nadrzędnego.
	0	navtree.1.order	Pozycja elementu w drzewie nawigacji map. Uwaga: Liczba w nazwie właściwości odnosi się do numeru elementu. Wartość właściwości odnosi się do pozycji elementu w drzewie nawigacji map. Pozycja elementu nadrzędnego jest określana w obrębie całego drzewa nawigacji map. Pozycja elementu podrzędnego jest określana w obrębie elementu nadrzędnego.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Drzewo nawigacji map*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Map navigation tree*

Skonfiguruj widżet *Map navigation tree*, który wyświetla następujące drzewo nawigacji map:

- Element A
  - Element A1
  - Element A2
- Element B
  - Element B1 (zawiera podlinkowaną mapę „1”, którą można wyświetlić w podlinkowanym widżecie *Map widget*)
  - Element B2
- Element C

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "navtree",
                        "name": "Map navigation tree",
                        "x": 0,
                        "y": 0,
                        "width": 18,
                        "height": 5,
                        "view_mode": 0,
                        "fields": [
                            {
                                "type": 1,
                                "name": "navtree.1.name",
                                "value": "Element A"
                            },
                            {
                                "type": 1,
                                "name": "navtree.2.name",
                                "value": "Element B"
                            },
                            {
                                "type": 1,
                                "name": "navtree.3.name",
                                "value": "Element C"
                            },
                            {
                                "type": 1,
                                "name": "navtree.4.name",
                                "value": "Element A1"
                            },
                            {
                                "type": 1,
                                "name": "navtree.5.name",
                                "value": "Element A2"
                            },
                            {
                                "type": 1,
                                "name": "navtree.6.name",
                                "value": "Element B1"
```

```
    },
    {
        "type": 1,
        "name": "navtree.7.name",
        "value": "Element 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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`
- Mapy

21 Wykres kołowy

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Wykres kołowy* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Wykres kołowy*, zapoznaj się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Wykres kołowy*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.

Zestaw danych

Poniższe parametry są obsługiwane podczas konfigurowania *zestawu danych*.

Note:
Pierwsza liczba w nazwie właściwości (np. ds.0.hosts.0, ds.0.items.0) oznacza konkretny zestaw danych, natomiast druga liczba, jeśli występuje, oznacza skonfigurowany host lub pozycję.

Parametr	typ	nazwa	wartość
Typ zestawu danych	0	ds.0.dataset_type	0 - Lista pozycji; 1 - *(domyślnie)* Wzorce pozycji.
Pozycje	4	ds.0.itemids.0	Identyfikator pozycji. Podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu należy ustawić tylko pozycje skonfigurowane na szablonie. Uwaga: Aby skonfigurować wiele pozycji, utwórz obiekt pola widgetu pulpitu nawigacyjnego dla każdej pozycji, zwiększając liczbę w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Lista pozycji”, a opcja *Pozycje (widget)* nie jest ustawiona
Pozycje (widget)	1	ds.0.itemids.0._reference	Zamiast identyfikatora pozycji: `ABCDE._itemid` - ustaw zgodny widget (z parametrem *Odwołanie* ustawionym na „ABCDE”) jako źródło danych dla pozycji. Uwaga: Aby skonfigurować wiele widgetów, utwórz obiekt pola widgetu pulpitu nawigacyjnego dla każdego widgetu, zwiększając liczbę w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Lista pozycji”, a opcja *Pozycje* nie jest ustawiona
Kolor	1	ds.0.color.0	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *wymagany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Lista pozycji”
Typ pozycji	0	ds.0.type.0	0 - *(domyślnie)* Normalna; 1 - Łączna. Wartość „Łączna” można ustawić tylko dla jednej pozycji na całym wykresie. Zachowanie parametru: - *obsługiwany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Lista pozycji”

Parametr	typ	nazwa	wartość
Wzorce hostów	1	ds.0.hosts.0	Nazwa lub wzorzec hosta (np. „Zabbix*”). Zachowanie parametru: - *wymagany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Wzorce pozycji”
Wzorce pozycji	1	ds.0.items.0	Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu. Nazwa lub wzorzec pozycji (np. „*: Liczba przetworzonych *wartości na sekundę”). Podczas konfigurowania widgetu na pulpicie nawigacyjnym szablonu należy ustawić tylko wzorce pozycji skonfigurowanych na szablonie. Zachowanie parametru: - *wymagany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Wzorce pozycji”
Kolor	1	ds.0.color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Wzorce pozycji”, a opcja *Paleta kolorów* nie jest ustawiona
Paleta kolorów	0	ds.0.color_palette	Indeks palety kolorów. Możliwe wartości mieszczą się w zakresie od 0 do 11. Wartość domyślna: 0. Zachowanie parametru: - *obsługiwany*, jeśli dla opcji *Typ zestawu danych* ustawiono wartość „Wzorce pozycji”, a opcja *Kolor* nie jest ustawiona
Funkcja agregacji	0	ds.0.aggregate_function	1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - *(domyślnie)* last.
Agregacja zestawu danych	0	ds.0.dataset_aggregation	0 *(domyślnie)* brak; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum. Zachowanie parametru: - *obsługiwany*, jeśli dla opcji *Typ pozycji* ustawiono wartość „Łączna”
Etykieta zestawu danych	1	ds.0.data_set_label	Dowolna wartość tekstowa. Wartość domyślna: "" (pusta).
Znaczniki pozycji			
Typ oceny	0	ds.0.item_tags_evaltype	0 *(domyślnie)* And/Or; 2 - Or.
Nazwa znacznika	1	ds.0.item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odwołuje się do kolejności znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *znaczników pozycji*

Parametr	typ	nazwa	wartość
Operator	0	ds.0.item_tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Liczba w nazwie właściwości odwołuje się do kolejności znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *znaczników pozycji*
Wartość znacznika	1	ds.0.item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odwołuje się do kolejności znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *znaczników pozycji*

Wyświetlanie opcji

Poniższe parametry są obsługiwane podczas konfigurowania *opcji wyświetlania*.

Parametr	type	name	value
Wybór danych historycznych	0	source	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy.
Rysuj	0	draw_type	0 - *(domyślnie)* Wykres kołowy; 1 - Wykres pierścieniowy.
Szerokość	0	width	20 - 20% promienia; 30 - 30% promienia; 40 - 40% promienia; 50 - *(domyślnie)* 50% promienia. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Rysuj* jest ustawiona na „Wykres pierścieniowy”
Szerokość obramowania	0	stroke	Dozwolone wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 0. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Rysuj* jest ustawiona na „Wykres pierścieniowy”
Pokaż wartość całkowitą	0	total_show	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Rysuj* jest ustawiona na „Wykres pierścieniowy”
Rozmiar	0	value_size_type	0 - *(domyślnie)* Automatycznie; 1 - Niestandardowy. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż wartość całkowitą* jest ustawiona na „Włączone”

Parametr	type	name	value
Rozmiar (wartość niestandardowego rozmiaru)	0	value_size	Dozwolone wartości mieszczą się w zakresie od 1 do 100. Wartość domyślna: 20. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż wartość całkowitą* jest ustawiona na „Włączone”
Miejsca dziesiętne	0	decimal_places	Dozwolone wartości mieszczą się w zakresie od 0 do 6. Wartość domyślna: 2. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż wartość całkowitą* jest ustawiona na „Włączone”
Jednostki (pole wyboru)	0	units_show	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż wartość całkowitą* jest ustawiona na „Włączone”
Jednostki (wartość)	1	units	Dowolna wartość tekstowa. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Jednostki* (pole wyboru) jest ustawiona na „Włączone”
Pogrubienie	0	value_bold	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż wartość całkowitą* jest ustawiona na „Włączone”
Kolor	1	value_color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż wartość całkowitą* jest ustawiona na „Włączone”
Odstęp między sektorami	0	space	Dozwolone wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 1.
Scal sektory mniejsze niż N% (pole wyboru)	0	merge	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Scal sektory mniejsze niż N% (wartość)	0	merge_percent	Dozwolone wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 1. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Scal sektory mniejsze niż N%* (pole wyboru) jest ustawiona na „Włączone”

Parametr	type	name	value
Scal sek-tory mniejsze niż N% (kolor)	1	merge_color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *obsługiwane*, jeśli opcja *Scal sektory mniejsze niż N%* (pole wyboru) jest ustawiona na „Włączone”

Okres czasu

Następujące parametry są obsługiwane przy konfigurowaniu *Okresu czasu*.

Parametr	type	name	value
Okres czasu	1	time_period._referenc	DASHBOARD._timeperiod - ustaw selektor okresu czasu pulpitu jako źródło danych; ABCDE._timeperiod - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na ”ABCDE”) jako źródło danych. Domyślnie: DASHBOARD._timeperiod Alternatywnie możesz ustawić okres czasu tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni czasu bezwzględnego (YYYY-MM-DD hh:mm:ss) lub względnego (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono time_period.to
Do	1	time_period.to	Prawidłowy ciąg czasu w składni czasu bezwzględnego (YYYY-MM-DD hh:mm:ss) lub względnego (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono time_period.from

Legenda

Poniższe parametry są obsługiwane podczas konfigurowania *Legendy*.

Parametr	typ	nazwa	wartość
Pokaż legendę	0	legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Pokaż wartość	0	legend_value	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Pokaż funkcję agregującą	0	legend_aggregation	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Wiersze	0	legend_lines_mode	0 - *(domyślnie)* Stała; 1 - Zmienna. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”

Parametr	typ	nazwa	wartość
Liczba wierszy/ Maksymalna liczba wierszy	0	legend_lines	Dozwolone wartości mieszczą się w zakresie od 1 do 10. Wartość domyślna: 1. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”
Liczba kolumn	0	legend_columns	Dozwolone wartości mieszczą się w zakresie od 1 do 4. Wartość domyślna: 4. Zachowanie parametru: - *obsługiwane*, jeśli opcja *Pokaż legendę* jest ustawiona na „Włączone”, a opcja *Pokaż wartość* jest ustawiona na „Wyłączone”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Pie chart*. Więcej informacji na temat konfiguracji pulpitu znajduje się w `dashboard.create`.

Konfigurowanie widżetu *Pie chart*

Skonfiguruj widżet *Pie chart* w następujący sposób:

- 2 zestawy danych dla łącznie 9 pozycji na 1 hoście.
- Pierwszy zestaw danych jest typu ”Item list” i składa się z 3 pozycji, z których wszystkie mają typ ”Normal” i są reprezentowane różnymi kolorami.
- Drugi zestaw danych jest typu ”Item patterns”, składa się z 6 pozycji, ma skonfigurowaną agregację dla każdej pozycji i jest reprezentowany przez niestandardowy kolor.
- Drugi zestaw danych ma również niestandardową etykietę zestawu danych.
- Dane na wykresie kołowym są wyświetlane jako wykres pierścieniowy z niestandardową szerokością oraz wartością całkowitą z jednostkami w środku.
- Dane na wykresie kołowym są wyświetlane i agregowane dla niestandardowego okresu czasu obejmującego ostatnie 3 godziny.
- Legenda wykresu kołowego wyświetla skonfigurowane pozycje w 4 wierszach.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

22 Hosty z problemami

Opis

Te parametry i możliwe wartości właściwości dla odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Problem hosts* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Problem hosts*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Hosty z problemami*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (Widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Wyklucz grupy hostów	2	exclude_groupids.0	ID grupy hostów. Uwaga: Aby wykluczyć wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z numerem zwiększanym w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty (Widżet/Pulpit)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor hostów pulpitu jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Problem	1	problem	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Istotność	0	severities.0	0 - Niesklasyfikowane; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z numerem zwiększanym w nazwie właściwości.
Tagi problemu			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*

Parametr	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemu*
Pokaż wycis-zone prob-lemy	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Ukryj grupy bez prob-lemów	0	hide_empty_groups	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wyświetlanie prob-lemów	0	ext_ack	0 - *(domyślnie)* Wszystkie; 1 - Tylko niepotwierdzone; 2 - Rozdzielone.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Problem hosts*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Problem hosts*

Skonfiguruj widżet *Problem hosts*, który wyświetla hosty z grup hostów „2” i „4”, mające problemy z nazwą zawierającą ciąg „CPU” oraz o następujących poziomach ważności: „Warning”, „Average”, „High”, „Disaster”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problemhosts",
```

```
                        "name": "Problem hosts",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

23 Problemy

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Problems* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Problems*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widget *Problems*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Pokaż	0	show	1 - *(domyślnie)* Ostatnie problemy; 2 - Historia; 3 - Problemy.
Grupy hostów	2	groupids.0	Identyfikator grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widgetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Grupy hostów (widget)	1	groupids._reference	Zamiast identyfikatora grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.

Parametr	type	name	value
Wyklucz grupy hostów	2	exclude_groupids.0	Identyfikator grupy hostów. Uwaga: Aby wykluczyć wiele grup hostów, utwórz obiekt pola widgetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosty	3	hostids.0	Identyfikator hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widgetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* musi być albo w ogóle nieskonfigurowany, albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosty (widget/dashboard)	1	hostids._reference	Zamiast identyfikatora hosta: `DASHBOARD.hostids` - ustaw dashboardowy selektor hostów jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Problem	1	problem	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Ważność	0	severities.0	0 - Niesklasyfikowane; 1 - Informacyjne; 2 - Ostrzeżenie; 3 - Średnie; 4 - Wysokie; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widgetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Znaczniki problemu			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa znacznika	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników problemu*

Parametr	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników problemu*
Wartość znacznika	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników problemu*
Pokaż znaczniki	0	show_tags	0 - *(domyślnie)* Brak; 1 - 1; 2 - 2; 3 - 3.
Nazwa znacznika (format)	0	tag_name_format	0 - *(domyślnie)* Pełna; 1 - Skrócona; 2 - Brak. Zachowanie parametru: - *obsługiwany*, jeśli parametr *Pokaż znaczniki* jest ustawiony na „1”, „2” lub „3”
Priorytet wyświetlania znaczników	1	tag_priority	Rozdzielana przecinkami lista znaczników. Zachowanie parametru: - *obsługiwany*, jeśli parametr *Pokaż znaczniki* jest ustawiony na „1”, „2” lub „3”
Pokaż dane operacyjne	0	show_opdata	0 - *(domyślnie)* Brak; 1 - Oddzielnie; 2 - Z nazwą problemu.
Pokaż wyciszone problemy	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Stan potwierdzenia	0	acknowledgement_status	0 - *(domyślnie)* Wszystkie; 1 - Niepotwierdzone; 2 - Potwierdzone.
Przeze mnie	0	acknowledged_by_me	0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Parametr	type	name	value
Sortuj wpisy według	0	sort_triggers	1 - Ważność (malejąco); 2 - Host (rosnąco); 3 - Czas (rosnąco); 4 - *(domyślnie)* Czas (malejąco); 13 - Ważność (rosnąco); 14 - Host (malejąco); 15 - Problem (rosnąco); 16 - Problem (malejąco). Dla wszystkich wartości z wyjątkiem „Czas (malejąco)” i „Czas (rosnąco)” parametr *Pokaż oś czasu* musi być ustawiony na „Wyłączone”. Wartości „Host (rosnąco)” i „Host (malejąco)” nie są obsługiwane podczas konfigurowania widgetu na dashboardzie szablonu.
Pokaż oś czasu	0	show_timeline	0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie parametru: - *obsługiwany*, jeśli parametr *Sortuj wpisy według* jest ustawiony na „Czas (malejąco)” lub „Czas (rosnąco)”
Wyróżnij cały wiersz	0	highlight_row	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Ten parametr nie jest obsługiwany w motywach o wysokim kontraście.
Pokaż wier-sze	0	show_lines	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 25.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikatowa w obrębie dashboardu, do którego należy widget. Zachowanie parametru: - *wymagany*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Problemy*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Problemy*

Skonfiguruj widżet *Problemy*, który wyświetla problemy dla grupy hostów „4”, spełniające następujące warunki:

- Problemy, które mają znacznik o nazwie „scope”, zawierający wartości „performance” lub „availability”, albo „capacity”.
- Problemy o następujących poziomach ważności: „Warning”, „Average”, „High”, „Disaster”.

Dodatkowo skonfiguruj widżet tak, aby wyświetlał znaczniki i dane operacyjne.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problems",
```

```
                "name": "Problems",
                "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- dashboard.create
- dashboard.update

24 Problemy według ważności

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Problemy według ważności* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Problemy według ważności*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Problemy według ważności*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (Widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wyklucz grupy hostów	2	exclude_groupids.0	ID grupy hostów. Uwaga: Aby wykluczyć wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z numerem zwiększanym w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Hosty (Widżet/Pulpit)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor hosta pulpitu jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Problem	1	problem	Nazwa zdarzenia problemu (bez rozróżniania wielkości liter, pełna nazwa lub jej część).
Ważność	0	severities.0	0 - Niesklasyfikowane; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średnia; 4 - Wysoka; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widżetu pulpitu dla każdej wartości z numerem zwiększanym w nazwie właściwości.
Tagi problemu			
Typ obliczania	0	evaltype	0 - *(domyślnie)* I/Lub; 2 - Lub.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów problemu*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów problemu*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów problemu*
Pokaż	0	show_type	0 - *(domyślnie)* Grupy hostów; 1 - Sumy. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu, a domyślnie jest ustawiony na "Sumy".

Parametr	type	name	value
Układ	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż* jest ustawione na ”Sumy”
Pokaż dane operacyjne	0	show_opdata	0 - *(domyślnie)* Brak; 1 - Osobno; 2 - Z nazwą problemu.
Pokaż ukryte problemy	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Ukryj grupy bez problemów	0	hide_empty_groups	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane* jeśli *Pokaż* jest ustawione na ”Grupy hostów” Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wyświetlanie problemów	0	ext_ack	0 - *(domyślnie)* Wszystkie; 1 - Tylko niepotwierdzone; 2 - Rozdzielone.
Pokaż oś czasu	0	show_timeline	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Problemy według ważności*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Problemy według ważności*

Skonfiguruj widżet *Problemy według ważności*, który wyświetla łączną liczbę problemów dla wszystkich grup hostów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "problemsbysv",
                        "name": "Problems by severity",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

25 Wykres punktowy

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Wykres punktowy* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Wykres punktowy*, zapoznaj się z opisanym w poniższych tabelach działaniem parametrów.

Parametry

Następujące parametry są obsługiwane dla widżetu *Scatter plot*.

Parametr	type	nazwa	wartość
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie dashboardu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Zestaw danych

Następujące parametry są obsługiwane przy konfigurowaniu *Zestawu danych*.

Note:
Pierwsza liczba w nazwie właściwości (np. ds.0.hosts.0, ds.0.items.0) oznacza konkretny zestaw danych, natomiast druga liczba, jeśli występuje, oznacza skonfigurowany host lub pozycję.

Parametr	type	name	value
Typ zestawu danych ***Zestaw danych: Lista pozycji***	0	ds.0.dataset_type	0 - Lista pozycji; 1 - *(domyślnie)* Wzorce pozycji.
Kolor	1	ds.0.color.0	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”
Oś X	4	ds.0.x_axis_itemids.0	ID pozycji. Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko pozycje skonfigurowane w szablonie. Uwaga: Aby skonfigurować wiele pozycji, utwórz obiekt pola widżetu pulpitu dla każdej pozycji z inkrementowanym numerem w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”, a *Oś X (Widżet)* nie jest ustawiona
Oś X (Widżet)	1	ds.0.x_axis_itemids.0	Zamiast ID pozycji: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na ”`ABCDE`”) jako źródło danych dla pozycji. Uwaga: Aby skonfigurować wiele widżetów, utwórz obiekt pola widżetu pulpitu dla każdego widżetu z inkrementowanym numerem w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”, a *Oś X* nie jest ustawiona

Parametr	type	name	value
Oś Y	4	ds.0.y_axis_itemids.0	ID pozycji. Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko pozycje skonfigurowane w szablonie. Uwaga: Aby skonfigurować wiele pozycji, utwórz obiekt pola widżetu pulpitu dla każdej pozycji z inkrementowanym numerem w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”, a *Oś Y (Widżet)* nie jest ustawiona
Oś Y (Widżet)	1	ds.0.y_axis_itemids.0	Zamiast ID pozycji: `ABCDE._itemid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na ”ABCDE”) jako źródło danych dla pozycji. Uwaga: Aby skonfigurować wiele widżetów, utwórz obiekt pola widżetu pulpitu dla każdego widżetu z inkrementowanym numerem w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Lista pozycji”, a *Oś Y* nie jest ustawiona
Nadpisz host	1	ds.0.override_hostid._reference_	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem ”Reference” ustawionym na ”ABCDE”) jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Zestaw danych: Wzorce pozycji			
Kolor	1	ds.0.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwany*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”, a *Paleta kolorów* nie jest ustawiona
Paleta kolorów	0	ds.0.color_palette	Indeks palety kolorów. Możliwy zakres wartości: 0-11. Domyślnie: 0. Zachowanie parametru: - *obsługiwany*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”, a *Kolor* nie jest ustawiony
Wzorce hostów	1	ds.0.hosts.0	Nazwa lub wzorzec hosta (np. ”Zabbix*”). Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji” Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Wzorce pozycji osi X	1	ds.0.x_axis_items.0	Nazwa lub wzorzec pozycji (np. ”*: Number of processed *values per second”). Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”
Wzorce pozycji osi Y	1	ds.0.y_axis_items.0	Nazwa lub wzorzec pozycji (np. ”*: Number of processed *values per second”). Podczas konfigurowania widżetu na pulpicie szablonu należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagany*, jeśli *Typ zestawu danych* jest ustawiony na ”Wzorce pozycji”
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z inkrementowanym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (Widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na ”`ABCDE`”) jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Nadpisz host	1	ds.0.override_hostid._reference	`ABCDE._hostid`- ustaw kompatybilny widżet (z parametrem ”Reference” ustawionym na ”`ABCDE`”) jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hosta pulpitu jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Tagi hosta			
Typ oceny	0	evaltype_host	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Nazwa tagu	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* przy konfigurowaniu *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* przy konfigurowaniu *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wartość tagu	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Liczba w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* przy konfigurowaniu *Tagów hosta* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Znacznik	0	ds.0.marker	0 - *(domyślnie)* Elipsa; 1 - Kwadrat; 2 - Trójkąt; 3 - Romb; 4 - Gwiazda; 5 - Krzyżyk.
Rozmiar znacznika	0	ds.0.marker_size	0 - *(domyślnie)* Mały (6px); 1 - Średni (9px); 2 - Duży (15px).
Przesunięcie czasu	1	ds.0.timeshift	Prawidłowy ciąg czasu (np. 3600, 1h itd.). Możesz używać sufiksów czasu. Dozwolone są również wartości ujemne. Domyślnie: "" (puste).
Interwał agregacji	1	ds.0.aggregate_interval	Prawidłowy ciąg czasu (np. 3600, 1h itd.). Możesz używać sufiksów czasu. Domyślnie: 15m.
Funkcja agregacji	0	ds.0.aggregate_function	0 - nieużywane; 1 - min; 2 - max; 3 - *(domyślnie)* avg; 4 - count; 5 - sum; 6 - first; 7 - last.

Opcje wyświetlania

Następujące parametry są obsługiwane przy konfigurowaniu *Opcji wyświetlania*.

Parametr	type	name	value
Wybór danych historycznych	0	source	0 - *(domyślnie)* Auto; 1 - Historia; 2 - Trendy.

Parametr	type	name	value
Nazwy hostów na etykietach	0	show_hostnames	0 - *(domyślnie)* Auto; 1 - Pokaż; 2 - Ukryj. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na dashboardzie szablonu.

Okres czasu

Następujące parametry są obsługiwane przy konfigurowaniu *Okresu czasu*.

Parametr	type	name	value
Okres czasu	1	time_period._reference	`DASHBOARD._timeperiod` — ustaw selektor okresu czasu pulpitu jako źródło danych; `ABCDE._timeperiod` — ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres czasu tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.to`
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres czasu* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.from`

Osie

Następujące parametry są obsługiwane do konfiguracji *Osi*.

Parametr	type	name	value
Oś X	0	x_axis	0 - Wyłączona; 1 - *(domyślnie)* Włączona.
Oś Y	0	y_axis	0 - *(domyślnie)* Wyłączona; 1 - Włączona.
Min	1	x_axis_min	Dowolna wartość liczbowa. Domyślnie: "" (puste).
		y_axis_min	
Max	1	x_axis_max	Dowolna wartość liczbowa. Domyślnie: "" (puste).
		y_axis_max	
Jednostki (typ)	0	x_axis_units	0 - *(domyślnie)* Automatycznie; 1 - Statyczne.
		y_axis_units	
Jednostki (wartość)	1	x_axis_static_units	Dowolna wartość tekstowa. Domyślnie: "" (puste).
		y_axis_static_units	

Legenda

Następujące parametry są obsługiwane przy konfigurowaniu *Legendy*.

Parametr	type	name	value
Pokaż legendę	0	legend	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Pokaż funkcję agregacji	0	legend_aggregation	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na „Włączone”
Wiersze	0	legend_lines_mode	0 - *(domyślnie)* Stałe; 1 - Zmienne. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na „Włączone”
Liczba wierszy/ Maksymalna liczba wierszy	0	legend_lines	Możliwe wartości z zakresu: 1-10. Domyślnie: 1. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na „Włączone”
Liczba kolumn	0	legend_columns	Możliwe wartości z zakresu: 1-4. Domyślnie: 4. Zachowanie parametru: - *obsługiwane*, jeśli *Pokaż legendę* jest ustawione na „Włączone”

Progi

Następujące parametry są obsługiwane do konfigurowania *Progów*.

Note:
Liczba w nazwie właściwości *Thresholds* (np. thresholds.0.color) odnosi się do miejsca progu na liście, posortowanej rosnąco. Jeśli jednak progi są skonfigurowane w innej kolejności, wartości zostaną posortowane rosnąco po zaktualizowaniu konfiguracji widżetu w frontendzie Zabbix (np. `"thresholds.0.threshold":"5"` → `"thresholds.0.threshold":"1"`; `"thresholds.1.threshold":"1"` → `"thresholds.1.threshold": "5"`).

Parametr	type	name	value
Interpolacja kolorów	0	interpolation	0 - *(domyślnie)* Wyłączona; 1 - Włączona.
Kolor	1	thresholds.0.color	Szesnastkowy kod koloru (np. FF0000).
Oś X	1	thresholds.0.x_axis_threshold	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. ”1d”, ”2w”, ”4K”, ”8G”).
Oś Y	1	thresholds.0.y_axis_threshold	Dowolna wartość liczbowa. Obsługiwane są sufiksy (np. ”1d”, ”2w”, ”4K”, ”8G”).

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Scatter plot*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Scatter plot*

Skonfiguruj widżet *Scatter plot* w następujący sposób:

- 5 zestawów danych typu „Wzorce pozycji”, każdy z domyślną agregacją (avg, 15m) i różnymi znacznikami.
- Legenda wyświetla skonfigurowane pozycje w jednym wierszu i dwóch kolumnach.
- Progi są skonfigurowane tak, aby kolor znacznika zmieniał się na czerwony, gdy jego wartość osiągnie 80 na dowolnej osi.

Żądanie:

```
{
    "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
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Zobacz także

- Pole widżetu pulpitu

- dashboard.create
- dashboard.update

26 Raport SLA

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Raport SLA* w metodach dashboard.create i dashboard.update.

Attention:
Właściwości fields widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Raport SLA*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widżet *Raport SLA*.

Parametr	typ	nazwa	wartość
Interwał odświeżania	0	rf_rate	0 - *(domyślnie)* Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
SLA	10	slaid.0	Identyfikator SLA. Zachowanie parametru: - *wymagany*
Usługa	9	serviceid.0	Identyfikator usługi.
Pokaż okresy	0	show_periods	Możliwe wartości mieszczą się w zakresie od 1 do 100. Wartość domyślna: 20.
Od	1	date_from	Prawidłowy ciąg daty w formacie YYYY-MM-DD. Obsługiwane są daty względne z modyfikatorami d, w, M, y (np. now, now/d, now/w-1w itd.).
Do	1	date_to	Prawidłowy ciąg daty w formacie YYYY-MM-DD. Obsługiwane są daty względne z modyfikatorami d, w, M, y (np. now, now/d, now/w-1w itd.).

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Raport SLA*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w dashboard.create.

Konfigurowanie widżetu *Raport SLA*

Skonfiguruj widżet *Raport SLA*, który wyświetla raport SLA dla usługi SLA „4”, usługi „2”, za okres ostatnich 30 dni.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
```

```
                "widgets": [
                    {
                        "type": "slareport",
                        "name": "SLA report",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

27 Informacje o systemie

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Informacje systemowe* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Informacje systemowe*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *System Information*.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - *(domyślnie)* 15 minut.
Show	0	info_type	0 - *(domyślnie)* Statystyki systemu; 1 - Węzły wysokiej dostępności.
Show software update check details	0	show_software_update_check_details	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *supported* jeśli `AllowSoftwareUpdateCheck` w konfiguracji serwer Zabbix jest włączone i *Show* jest ustawione na ”System stats”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Informacje o systemie*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Informacje o systemie*

Skonfiguruj widżet *Informacje o systemie*, który wyświetla statystyki systemowe z interwałem odświeżania wynoszącym 10 minut oraz z włączonym sprawdzaniem aktualizacji oprogramowania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "systeminfo",
                        "name": "System information",
```

```
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

28 Najważniejsze hosty

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Top hosts* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Top hosts*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widget *Najlepsze hosty*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	Identyfikator grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widgetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Grupy hostów (widget)	1	groupids._reference	Zamiast identyfikatora grupy hostów: `ABCDE._hostgroupids` - ustaw zgodny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosty	3	hostids.0	Identyfikator hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widgetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* musi być całkowicie nieskonfigurowany albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosty (widget/dashboard)	1	hostids._reference	Zamiast identyfikatora hosta: `DASHBOARD.hostids` - ustaw dashboardowy selektor hostów jako źródło danych dla hostów; `ABCDE._hostids` - ustaw zgodny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Tagi hostów			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.

Parametr	type	name	value
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Operator	0	tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Pokaż hosty w konserwacji	0	maintenance	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Kolumny (patrz poniżej)			
Sortuj według	0	column	Wartość liczbowa kolumny spośród skonfigurowanych kolumn.
Kolejność	0	order	2 - *(domyślnie)* Najlepsze N; 3 - Najgorsze N.
Limit hostów	0	show_lines	Dozwolone wartości mieszczą się w zakresie od 1 do 1000. Wartość domyślna: 10. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.

Kolumny

Kolumny posiadają wspólne parametry oraz dodatkowe parametry w zależności od konfiguracji parametru *Data*.

Note:
Dla wszystkich parametrów związanych z kolumnami, liczba w nazwie właściwości (np. columns.name.0) odnosi się do kolumny, dla której parametr jest skonfigurowany.

Następujące parametry są obsługiwane dla wszystkich kolumn.

Parametr	type	name	value
Nazwa	1	columns.0.name	Dowolna wartość tekstowa. Zachowanie parametru: - *wymagany*
Dane	0	columns.0.data	1 - Wartość pozycji; 2 - Nazwa hosta; 3 - Tekst. Zachowanie parametru: - *wymagany*
Kolor bazowy	1	columns.0.base_color	Szesnastkowy kod koloru (np. FF0000). Zachowanie parametru: - *wymagany*

Wartość pozycji

Poniższe parametry są obsługiwane, jeśli dla *Data* ustawiono wartość „Item value”.

Note:
Pierwsza liczba w nazwie właściwości *Thresholds* (np. columnsthresholds.0.color.0) odwołuje się do kolumny, dla której skonfigurowano progi, natomiast druga liczba odwołuje się do miejsca progu na liście posortowanej rosnąco. Jeśli jednak progi zostaną skonfigurowane w innej kolejności, po zaktualizowaniu konfiguracji widżetu w frontend Zabbix wartości zostaną posortowane rosnąco (np. `"threshold.0.threshold":"5"` → `"threshold.0.threshold":"1"`; `"threshold.1.threshold":"1"` → `"threshold.1.threshold": "5"`).

Parametr	type	name	value
Pozycja	1	columns.0.item	Prawidłowa nazwa pozycji. Podczas konfigurowania widżetu na pulpicie nawigacyjnym szablonu należy ustawić wyłącznie pozycje skonfigurowane na szablonie.
Wyświetlaj wartość pozycji jako	0	columns.0.display_value_as	0 - *(domyślnie)* Liczbowa; 1 - Tekstowa; 2 - Binarna.
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Bez zmian; 2 - Pasek; 3 - Wskaźniki; 6 - Sparkline. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”
Min	1	columns.0.min	Dowolna wartość liczbowa. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Bar” lub „Indicators”

Parametr	type	name	value
Maks.	1	columns.0.max	Dowolna wartość liczbowa. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Bar” lub „Indicators”
Miejsca dziesiętne	0	columns.0.decimal_places	Możliwe wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 2. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”
Sparkline			
Szerokość	0	columns.0.sparkline.width	Możliwe wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 1. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline”
Wypełnienie	0	columns.0.sparkline.fill	Możliwe wartości mieszczą się w zakresie od 0 do 10. Wartość domyślna: 3. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline”
Kolor	1	columns.0.sparkline.color	Szesnastkowy kod koloru (np. `FF0000`). Wartość domyślna: `42A5F5`. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline”
Okres	1	columns.0.sparkline.time_period.from	`DASHBOARD._timeperiod` - ustaw selektor okresu pulpitu nawigacyjnego jako źródło danych; `ABCDE._timeperiod` - ustaw zgodny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Wartość domyślna: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres wyłącznie za pomocą parametrów *Od* i *Do*. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline”
Od	1	columns.0.sparkline.time_period.from	Prawidłowy znacznik czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline” - *wymagane*, jeśli ustawiono `columns.0.sparkline.time_period.to`

Parametr	type	name	value
Do	1	columns.0.sparkline.time_period.to	Prawidłowy czas czy skład bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline” - *wymagane*, jeśli ustawiono `columns.0.sparkline.time_period.from`
Dane historyczne	0	columns.0.sparkline.history	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”, a dla *Wyświetlanie* ustawiono „Sparkline”
Progi			
Kolor	1	columnsthresholds.0.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”
Próg	1	columnsthresholds.0.threshold	Dowolna wartość tekstowa. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”
Wyróżnienia			
Kolor	1	columns.0.highlights.0.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Text”
Wzorzec	1	columns.0.highlights.0.pattern	Dowolna wartość tekstowa. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Text”
Pokaż miniaturę	0	columns.0.show_thumbnail	0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Binary”
Funkcja agregu-jąca	0	columns.0.aggregate_function	0 - *(domyślnie)* nieużywana; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last.

Parametr	type	name	value
Okres	1	columns.0.time_period.from	DASHBOARD. _timeperiod - ustaw selektor okresu pulpitu nawigacyjnego jako źródło danych; ABCDE._timeperiod - ustaw zgodny widżet (z parametrem reference równym ABCDE) jako źródło danych. Wartość domyślna: DASHBOARD._timeperiod Alternatywnie można ustawić okres wyłącznie za pomocą parametrów *Od* i *Do*. Zachowanie parametru: - *obsługiwane*, jeśli dla *Funkcja agregująca* ustawiono „min”, „max”, „avg”, „count”, „sum”, „first” lub „last”
Od	1	columns.0.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub względnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwane*, jeśli nie ustawiono *Okres*, a dla *Funkcja agregująca* ustawiono „min”, „max”, „avg”, „count”, „sum”, „first” lub „last” - *wymagane*, jeśli ustawiono columns.0.time_period.to
Do	1	columns.0.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (YYYY-MM-DD hh:mm:ss) lub względnej (now, now/d, now/w-1w itd.). Zachowanie parametru: - *obsługiwane*, jeśli nie ustawiono *Okres*, a dla *Funkcja agregująca* ustawiono „min”, „max”, „avg”, „count”, „sum”, „first” lub „last” - *wymagane*, jeśli ustawiono columns.0.time_period.from
Dane historyczne	0	columns.0.history	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy. Zachowanie parametru: - *obsługiwane*, jeśli dla *Wyświetlaj wartość pozycji jako* ustawiono „Numeric”
Odwołanie	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. ABCDE lub JBPNL). Wartość musi być unikatowa w obrębie pulpitu nawigacyjnego, do którego należy widżet. Zachowanie parametru: - *wymagane*

Tekst

Następujące parametry są obsługiwane, jeśli *Data* jest ustawione na ”Text”.

Parameter	type	name	value
Text	1	columns.0.text	Dowolna wartość tekstowa, w tym makra. Obsługiwane makra: {HOST.*}, {INVENTORY.*}. Zachowanie parametru: - *wymagane*, jeśli *Data* jest ustawione na ”Text”

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Top hosts*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w dashboard.create.

Konfigurowanie widżetu *Top hosts*

Skonfiguruj widżet *Top hosts*, który wyświetla hosty o najwyższym wykorzystaniu CPU w grupie hostów „4”. Dodatkowo skonfiguruj następujące niestandardowe kolumny: „Nazwa hosta”, „Wykorzystanie CPU w %”, „Średnia 1m”, „Średnia 5m”, „Średnia 15m”, „Procesy”.

Żądanie:

```
{
    "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
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```


Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

29 Najważniejsze pozycje

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Top items* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Top items*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Top items*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z numerem zwiększanym w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z numerem zwiększanym w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty (widżet/pulpit)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor hostów pulpitu jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Tagi hostów			
Typ obliczania	0	host_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Nazwa tagu	1	host_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Operator	0	host_tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wartość tagu	1	host_tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów hostów* Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Układ	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.
Pokaż problemy	0	show_problems	0 - Wszystkie; 1 - *(domyślnie)* Nieukryte; 2 - Brak.
Pozycje (zobacz poniżej)			

Zaawansowana konfiguracja

Następujące zaawansowane parametry konfiguracji są obsługiwane przez widżet *Najważniejsze pozycje*.

Parametr	type	name	value
Sortowanie hostów			
Sortuj według	0	host_ordering_order_by	0 - *(domyślnie)* Nazwa hosta; 3 - Wartość pozycji.
Wzorce pozycji	1	host_ordering_item.0	Nazwa lub wzorzec pozycji (np. ”*: Liczba przetworzonych *wartości na sekundę”). Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu dashboardu dla każdego wzorca pozycji, zwiększając numer w nazwie właściwości. Podczas konfigurowania widżetu na dashboardzie szablonu należy ustawić tylko wzorce dla pozycji skonfigurowanych w szablonie. Zachowanie parametru: - *wymagany*, jeśli opcja *Sortuj według* jest ustawiona na „Wartość pozycji”
Kolejność	0	host_ordering_order	2 - *(domyślnie)* Pierwsze N; 3 - Ostatnie N.

Parametr	type	name	value
Limit	0	host_ordering_limit	Dozwolone wartości mieszczą się w zakresie od 1 do 100. Wartość domyślna: 10.
Sortowanie pozycji			
Sortuj według	0	item_ordering_order_by	1 - Host; 2 - Nazwa pozycji; 3 - *(domyślnie)* Wartość pozycji.
Wzorce hostów	1	item_ordering_host.0	Nazwa lub wzorzec hosta. Uwaga: Aby skonfigurować wiele wzorców hostów, utwórz obiekt pola widżetu dashboardu dla każdego wzorca hosta, zwiększając numer w nazwie właściwości. Zachowanie parametru: - *wymagany*, jeśli opcja *Sortuj według* jest ustawiona na „Host”
Kolejność	0	item_ordering_order	2 - *(domyślnie)* Pierwsze N; 3 - Ostatnie N.
Limit	0	item_ordering_limit	Dozwolone wartości mieszczą się w zakresie od 1 do 100. Wartość domyślna: 10.

Kolumny

Note:
Pierwsza liczba w nazwie właściwości (np. columns.0.items.0, columns.0.item_tags_evaltype) oznacza konkretną kolumnę, natomiast druga liczba, jeśli występuje, oznacza skonfigurowaną encję (np. wzorzec pozycji, znacznik).

Parametr	type	name	value
Wzorce pozycji	1	columns.0.items.0	Nazwa lub wzorzec pozycji (np. ”*: Liczba przetworzonych *wartości na sekundę”). Uwaga: Aby skonfigurować wiele wzorców pozycji, utwórz obiekt pola widżetu dashboardu dla każdego wzorca pozycji, zwiększając drugą liczbę w nazwie właściwości. Podczas konfigurowania widżetu na dashboardzie szablonu należy ustawić tylko wzorce pozycji skonfigurowanych na szablonie. Zachowanie parametru: - *wymagany*
Znaczniki pozycji			
Typ oceny	0	columns.0.item_tags_evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa znacznika	1	columns.0.item_tags.0.tag	Dowolna wartość tekstowa. Uwaga: Druga liczba w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników pozycji*

Parametr	type	name	value
Operator	0	columns.0.item_tags.0.operator	**0** - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Druga liczba w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników pozycji*
Wartość znacznika	1	columns.0.item_tags.0.value	Dowolna wartość tekstowa. Uwaga: Druga liczba w nazwie właściwości wskazuje kolejność znacznika na liście oceny znaczników. Zachowanie parametru: - *wymagany* podczas konfigurowania *Znaczników pozycji*
Kolor bazowy	1	columns.0.base_color	Szesnastkowy kod koloru (np. FF0000).
Wyświetlaj wartość jako	0	columns.0.display_value_as	1 - *(domyślnie)* Liczba; 2 - Tekst.
Wyświetlanie	0	columns.0.display	1 - *(domyślnie)* Bez zmian; 2 - Pasek; 3 - Wskaźniki; 6 - Sparkline.
Min.	1	columns.0.min	Dowolna wartość liczbowa. Obsługiwane są przyrostki (np. ”1d”, ”2w”, ”4K”, ”8G”). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”, a *Wyświetlanie* na „Pasek” lub „Wskaźniki”
Maks.	1	columns.0.max	Dowolna wartość liczbowa. Obsługiwane są przyrostki (np. ”1d”, ”2w”, ”4K”, ”8G”). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”, a *Wyświetlanie* na „Pasek” lub „Wskaźniki”
Sparkline			
Szerokość	0	columns.0.sparkline.width	Możliwe wartości mieszczą się w zakresie od 0 do 10. Domyślnie: 1. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline”
Wypełnienie	0	columns.0.sparkline.fill	Możliwe wartości mieszczą się w zakresie od 0 do 10. Domyślnie: 3. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline”
Kolor	1	columns.0.sparkline.color	Szesnastkowy kod koloru (np. FF0000). Domyślnie: 42A5F5. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline”

Parametr	type	name	value
Okres	1	columns.0.sparkline.time_period.from	`DASHBOARD._timeperiod` - ustaw selektor okresu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw zgodny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres tylko w parametrach *Od* i *Do*. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline”
Od	1	columns.0.sparkline.time_period.from	Prawidłowy zakres czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline” - *wymagany*, jeśli ustawiono `columns.0.sparkline.time_period.to`
Do	1	columns.0.sparkline.time_period.to	Prawidłowy zakres czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline” - *wymagany*, jeśli ustawiono `columns.0.sparkline.time_period.from`
Dane historyczne	0	columns.0.sparkline.history	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlanie* jest ustawione na „Sparkline”
Progi			
Kolor	1	columns.0.thresholds.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”
Próg	1	columns.0.thresholds.threshold	Dowolna wartość liczbowa. Obsługiwane są przyrostki (np. ”1d”, ”2w”, ”4K”, ”8G”). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”
Wyróżnienia			
Wyróżnienie	1	columns.0.highlights.color	Szesnastkowy kod koloru (np. `FF0000`). Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Tekst”
Próg	1	columns.0.highlights.pattern	Dowolne wyrażenie regularne. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Tekst”
Miejsca dziesiętne	0	columns.0.decimal_places	Możliwe wartości mieszczą się w zakresie od 0 do 10. Domyślnie: 2.
Konfiguracja zaawansowana			

Parametr	type	name	value
Funkcja agregacji	0	columns.0.aggregate_function	Jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”: 0 - *(domyślnie)* nieużywana; 1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum; 6 - first; 7 - last. Jeśli *Wyświetlaj wartość jako* jest ustawione na „Tekst”: 0 - *(domyślnie)* nieużywana; 4 - count; 6 - first; 7 - last.
Okres	1	columns.0.time_period.from columns.0.time_period.to	`DASHBOARD._timeperiod` - ustaw selektor okresu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw zgodny widżet (z parametrem `reference` równym `ABCDE`) jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie można ustawić okres tylko w parametrach *Od* i *Do*. Zachowanie parametru: - *obsługiwany*, jeśli *Funkcja agregacji* jest ustawiona na „min”, „max”, „avg”, „count”, „sum”, „first” lub „last”
Od	1	columns.0.time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres* nie jest ustawiony, a *Funkcja agregacji* jest ustawiona na „min”, „max”, „avg”, „count”, „sum”, „first” lub „last” - *wymagany*, jeśli ustawiono `columns.0.time_period.to`
Do	1	columns.0.time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli *Okres* nie jest ustawiony, a *Funkcja agregacji* jest ustawiona na „min”, „max”, „avg”, „count”, „sum”, „first” lub „last” - *wymagany*, jeśli ustawiono `columns.0.time_period.from`
Dane historyczne	0	columns.0.history	0 - *(domyślnie)* Automatycznie; 1 - Historia; 2 - Trendy. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”
Agreguj	0	columns.0.aggregate_grouping	0 - *(domyślnie)* Każda pozycja 1 - Połączone. Zachowanie parametru: - *obsługiwany*, jeśli *Wyświetlaj wartość jako* jest ustawione na „Liczba”, a *Wyświetlanie* na „Bez zmian”, „Pasek” lub „Wskaźniki”.
Funkcja agregacji połączonych	0	columns.0.column_aggregate_function	1 - min; 2 - max; 3 - avg; 4 - count; 5 - sum *(domyślnie)*. Zachowanie parametru: - *wymagany* podczas konfigurowania *Agreguj*.

Parametr	type	name	value
Nazwa połączonej kolumny	1	columns.0.combined	Dowolna wartość tekstowa. Zachowanie parametru: - *wymagany* podczas konfigurowania *Agreguj*.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widgetu pulpitu dla widgetu *Top items*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Top items*

Skonfiguruj widżet *Top items*, który wyświetla dane dla hosta „10084" i tylko dla pozycji, dla których tag o nazwie „component" zawiera wartość „cpu". Dodatkowo wyświetl dane z hostami umieszczonymi u góry i użyj kolorowego paska wskaźnika do reprezentacji komórek.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

30 najważniejszych wyzwalaczy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widgetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widgetu *Najczęstsze wyzwalacze* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Najczęstsze wyzwalacze*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Poniższe parametry są obsługiwane przez widget *Top triggers*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - *(domyślnie)* Brak odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	Identyfikator grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widgetu dashboardu dla każdej grupy hostów, zwiększając numer w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Hosty	3	hostids.0	Identyfikator hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widgetu dashboardu dla każdego hosta, zwiększając numer w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* musi być całkowicie nieskonfigurowany albo skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widgetu na dashboardzie szablonu.
Problem	1	problem	Nazwa zdarzenia problemu (bez uwzględniania wielkości liter, pełna nazwa lub jej część).
Ważność	0	severities.0	Poziomy ważności wyzwalacza. 0 - Niesklasyfikowane; 1 - Informacyjne; 2 - Ostrzeżenie; 3 - Średnie; 4 - Wysokie; 5 - Katastrofa. Domyślnie: puste (wszystkie włączone). Uwaga: Aby skonfigurować wiele wartości, utwórz obiekt pola widgetu dashboardu dla każdej wartości, zwiększając numer w nazwie właściwości.
Tagi problemu			
Typ oceny	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów problemu*

Parametr	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów problemu*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości wskazuje kolejność tagu na liście oceny tagów. Zachowanie parametru: - *wymagany* podczas konfigurowania *Tagów problemu*
Okres	1	time_period._reference	`DASHBOARD._timeperiod` - ustaw selektor okresu dashboardu jako źródło danych; `ABCDE._timeperiod` - ustaw kompatybilny widget (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych. Domyślnie: `DASHBOARD._timeperiod` Alternatywnie możesz ustawić okres tylko w parametrach *Od* i *Do*.
Od	1	time_period.from	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli parametr *Okres* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.to`
Do	1	time_period.to	Prawidłowy ciąg czasu w składni bezwzględnej (`YYYY-MM-DD hh:mm:ss`) lub względnej (`now`, `now/d`, `now/w-1w` itd.). Zachowanie parametru: - *obsługiwany*, jeśli parametr *Okres* nie jest ustawiony - *wymagany*, jeśli ustawiono `time_period.from`
Limit wyzwalaczy	0	show_lines	Możliwe wartości mieszczą się w zakresie od 1 do 100. Domyślnie: 10.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Top triggers*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Top triggers*

Skonfiguruj widżet *Top triggers*, który wyświetla 5 najważniejszych wyzwalaczy dla grupy hostów „4" wraz z liczbą wszystkich problemów dla każdego wyzwalacza. Widżet wyświetla tylko wyzwalacze o poziomach ważności „Warning", „Average", „High" lub „Disaster" oraz problemy, które mają tag o nazwie „scope", zawierający wartości „performance", „availability" lub „capacity".

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

31 Przegląd wyzwalaczy

Opis

Te parametry oraz możliwe wartości właściwości dla odpowiednich obiektów pól widżetu pulpitu nawigacyjnego umożliwiają skonfigurowanie widżetu *Przegląd wyzwalaczy* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widżetu nie są sprawdzane podczas tworzenia ani aktualizowania pulpitu nawigacyjnego. Umożliwia to użytkownikom modyfikowanie wbudowanych widżetów i tworzenie niestandardowych widżetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widżetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widżetu *Przegląd wyzwalaczy*, zapoznaj się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Przegląd wyzwalaczy*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Pokaż	0	show	1 - *(domyślnie)* Ostatnie problemy; 2 - Dowolne; 3 - Problemy.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z inkrementowanym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (Widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z inkrementowanym numerem w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty (Widżet/Pulpit)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor hostów pulpitu jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Parametr	type	name	value
Tagi problemów			
Typ oceny	0	evaltype	0 - *(domyślnie)* I/Lub; 2 - Lub.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemów*
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemów*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście oceny tagów. Zachowanie parametru: - *wymagane* podczas konfigurowania *Tagów problemów*
Pokaż wycisz one problemy	0	show_suppressed	0 - *(domyślnie)* Wyłączone; 1 - Włączone.
Układ	0	layout	0 - *(domyślnie)* Poziomy; 1 - Pionowy.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Przegląd wyzwalaczy*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Przegląd wyzwalaczy*

Skonfiguruj widżet *Przegląd wyzwalaczy*, który wyświetla stany wyzwalaczy dla wszystkich grup hostów mających wyzwalacze z tagiem o nazwie „scope” i zawierającym wartość „availability”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- dashboard.create
- dashboard.update

32 URL

Opis

Te parametry oraz możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *URL* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *URL*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *URL*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - *(domyślnie)* Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
URL	1	url	Prawidłowy ciąg URL. Zachowanie parametru: - *wymagany*
Nadpisz host	1	override_hostid._reference	`ABCDE._hostid` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "ABCDE") jako źródło danych dla hostów; `DASHBOARD._hostid` - ustaw selektor hosta jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *URL*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *URL*

Skonfiguruj widżet *URL*, który wyświetla stronę główną podręcznika Zabbix.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

33 Monitorowanie WWW

Opis

Te parametry i możliwe wartości właściwości odpowiednich obiektów pól widgetu dashboardu umożliwiają skonfigurowanie widgetu *Monitorowanie sieci WWW* w metodach `dashboard.create` i `dashboard.update`.

Attention:
Właściwości `fields` widgetu nie są sprawdzane podczas tworzenia ani aktualizowania dashboardu. Umożliwia to użytkownikom modyfikowanie wbudowanych widgetów i tworzenie niestandardowych widgetów, ale wiąże się również z ryzykiem nieprawidłowego utworzenia lub zaktualizowania widgetów. Aby zapewnić pomyślne utworzenie lub zaktualizowanie widgetu *Monitorowanie sieci WWW*, należy zapoznać się z opisem działania parametrów przedstawionym w poniższych tabelach.

Parametry

Następujące parametry są obsługiwane dla widżetu *Monitorowanie WWW*.

Parametr	type	name	value
Interwał odświeżania	0	rf_rate	0 - Bez odświeżania; 10 - 10 sekund; 30 - 30 sekund; 60 - *(domyślnie)* 1 minuta; 120 - 2 minuty; 600 - 10 minut; 900 - 15 minut.
Grupy hostów	2	groupids.0	ID grupy hostów. Uwaga: Aby skonfigurować wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z inkrementowanym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Grupy hostów (Widżet)	1	groupids._reference	Zamiast ID grupy hostów: `ABCDE._hostgroupids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla grup hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Wyklucz grupy hostów	2	exclude_groupids.0	ID grupy hostów. Uwaga: Aby wykluczyć wiele grup hostów, utwórz obiekt pola widżetu pulpitu dla każdej grupy hostów z inkrementowanym numerem w nazwie właściwości. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty	3	hostids.0	ID hosta. Uwaga: Aby skonfigurować wiele hostów, utwórz obiekt pola widżetu pulpitu dla każdego hosta z inkrementowanym numerem w nazwie właściwości. W przypadku wielu hostów parametr *Grupy hostów* nie może być w ogóle skonfigurowany albo musi być skonfigurowany z co najmniej jedną grupą hostów, do której należą skonfigurowane hosty. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Hosty (Widżet/Pulpit)	1	hostids._reference	Zamiast ID hosta: `DASHBOARD.hostids` - ustaw selektor hostów pulpitu jako źródło danych dla hostów; `ABCDE._hostids` - ustaw kompatybilny widżet (z parametrem *Reference* ustawionym na "`ABCDE`") jako źródło danych dla hostów. Ten parametr nie jest obsługiwany podczas konfigurowania widżetu na pulpicie szablonu.
Tagi scenariusza			
Typ obliczania	0	evaltype	0 - *(domyślnie)* And/Or; 2 - Or.
Nazwa tagu	1	tags.0.tag	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów scenariusza*

Parametr	type	name	value
Operator	0	tags.0.operator	0 - Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów scenariusza*
Wartość tagu	1	tags.0.value	Dowolna wartość tekstowa. Uwaga: Numer w nazwie właściwości odnosi się do kolejności tagu na liście obliczania tagów. Zachowanie parametru: - *wymagane* przy konfigurowaniu *Tagów scenariusza*
Pokaż hosty w utrzymaniu	0	maintenance	0 - Wyłączone; 1 - *(domyślnie)* Włączone.
Reference	1	reference	Dowolna wartość tekstowa składająca się z 5 znaków (np. `ABCDE` lub `JBPNL`). Ta wartość musi być unikalna w obrębie pulpitu, do którego należy widżet. Zachowanie parametru: - *wymagane*

Przykłady

Poniższe przykłady mają na celu jedynie opisanie konfiguracji obiektów pól widżetu pulpitu dla widżetu *Web monitoring*. Więcej informacji na temat konfiguracji pulpitu można znaleźć w `dashboard.create`.

Konfigurowanie widżetu *Web monitoring*

Skonfiguruj widżet *Web monitoring*, który wyświetla podsumowanie statusu aktywnych scenariuszy monitorowania WWW dla grupy hostów „4”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dashboard.create",
    "params": {
        "name": "My dashboard",
        "display_period": 30,
        "auto_start": 1,
        "pages": [
            {
                "widgets": [
                    {
                        "type": "web",
                        "name": "Web monitoring",
                        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pole widżetu pulpitu
- `dashboard.create`
- `dashboard.update`

Pulpit szablonu

Ta klasa została zaprojektowana do pracy z pulpitami szablonów.

Odwołania do obiektów:

- Pulpit szablonu
- Strona pulpitu szablonu
  - Widget pulpitu szablonu
    - Pole widgetu pulpitu szablonu

Dostępne metody:

- templatedashboard.create - utwórz nowe pulpity szablonów
- templatedashboard.delete - usuń pulpity szablonów
- templatedashboard.get - pobierz pulpity szablonów
- templatedashboard.update - zaktualizuj pulpity szablonów

Obiekt pulpitu nawigacyjnego szablonu

Następujące obiekty są bezpośrednio powiązane z API `templateashboard`.

Dashboard szablonu

Obiekt dashboardu szablonu ma następujące właściwości.

Property	Type	Description
dashboardid	ID	ID dashboardu szablonu. Property behavior: - *read-only* - *required* for update operations
name	string	Nazwa dashboardu szablonu. Property behavior: - *required* for create operations
templateid	ID	ID szablonu, do którego należy dashboard. Property behavior: - *constant* - *required* for create operations
display_period	integer	Domyślny interwał pokazu slajdów (w sekundach). Possible values: 10, 30, 60, 120, 600, 1800, 3600. Default: 30.
auto_start	integer	Automatyczne uruchamianie pokazu slajdów. Possible values: 0 - nie uruchamiaj automatycznie pokazu slajdów; 1 - *(default)* uruchamiaj automatycznie pokaz slajdów.
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia zaimportowanych dashboardów szablonów z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany.

Strona dashboardu szablonu

Obiekt strony dashboardu szablonu ma następujące właściwości.

Property	Type	Description
dashboard_pageid	ID	ID strony dashboardu. Zachowanie właściwości: - *tylko do odczytu*
name	string	Nazwa strony dashboardu. Domyślnie: pusty ciąg znaków.
display_period	integer	Interwał pokazu slajdów dashboardu (w sekundach). Możliwe wartości: 0, 10, 30, 60, 120, 600, 1800, 3600. Domyślnie: 0 (zostanie użyty domyślny interwał pokazu slajdów).
widgets	array	Tablica obiektów widgetu dashboardu szablonu.

Widżet pulpitu szablonu

Obiekt widżetu pulpitu szablonu ma następujące właściwości.

Właściwość	Typ	Opis
widgetid	ID	ID widżetu pulpitu. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
type	string	Typ widżetu pulpitu. Możliwe wartości: actionlog - Dziennik akcji; clock - Zegar; discovery - Status wykrywania; favgraphs - Ulubione wykresy; favmaps - Ulubione mapy; gauge - Wskaźnik; graph - Wykres (klasyczny); graphprototype - Prototyp wykresu; honeycomb - Honeycomb; hostavail - Dostępność hosta; hostcard - Karta hosta; hostnavigator - Nawigator hostów; itemcard - Karta pozycji; itemnavigator - Nawigator pozycji; item - Wartość pozycji; map - Mapa; navtree - Drzewo nawigacji map; piechart - Wykres kołowy; plaintext - Zwykły tekst; problemhosts - Hosty z problemami; problems - Problemy; problemsbysv - Problemy według ważności; scatterplot - Wykres punktowy; slareport - Raport SLA; svggraph - Wykres; systeminfo - Informacje o systemie; tophosts - Najlepsze hosty; topitems - Najlepsze pozycje; toptriggers - Najważniejsze wyzwalacze; trigover - Przegląd wyzwalaczy; url - URL; web - Monitorowanie WWW. Zachowanie właściwości: - *wymagane*
name	string	Niestandardowa nazwa widżetu.
x	integer	Pozycja pozioma od lewej strony pulpitu. Możliwe wartości mieszczą się w zakresie od 0 do 71.
y	integer	Pozycja pionowa od górnej krawędzi pulpitu. Możliwe wartości mieszczą się w zakresie od 0 do 63.
width	integer	Szerokość widżetu. Możliwe wartości mieszczą się w zakresie od 1 do 72.
height	integer	Wysokość widżetu. Możliwe wartości mieszczą się w zakresie od 1 do 64.
view_mode	integer	Tryb widoku widżetu. Możliwe wartości: 0 - *(domyślnie)* domyślny widok widżetu; 1 - z ukrytym nagłówkiem;
fields	array	Tablica obiektów pola widżetu pulpitu szablonu.

Pole widgetu pulpitu szablonu

Obiekt pola widgetu pulpitu szablonu ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Typ pola widgetu. Możliwe wartości: 0 - Integer; 1 - String; 4 - Pozycja; 5 - Prototyp pozycji; 6 - Wykres; 7 - Prototyp wykresu; 8 - Mapa; 9 - Usługa; 10 - SLA; 11 - Użytkownik; 12 - Akcja; 13 - Typ nośnika. Zachowanie właściwości: - *wymagane*
name	string	Nazwa pola widgetu. Możliwe wartości: patrz Pola widgetu pulpitu. Należy pamiętać, że niektóre parametry związane z hostem (np. *Grupy hostów*, *Wyklucz grupy hostów* i *Hosty* w widżecie *Problemy*, *Grupy hostów* w widżecie *Dostępność hostów* itd.) nie są dostępne podczas konfigurowania widgetu na pulpicie szablonu. Wynika to z tego, że pulpity szablonów wyświetlają dane tylko z hosta, do którego szablon jest podłączony. Zachowanie właściwości: - *wymagane*
value	mixed	Wartość pola widgetu zależna od typu. Możliwe wartości: patrz Pola widgetu pulpitu. Należy pamiętać, że niektóre parametry związane z hostem (np. *Grupy hostów*, *Wyklucz grupy hostów* i *Hosty* w widżecie *Problemy*, *Grupy hostów* w widżecie *Dostępność hostów* itd.) nie są dostępne podczas konfigurowania widgetu na pulpicie szablonu. Wynika to z tego, że pulpity szablonów wyświetlają dane tylko z hosta, do którego szablon jest podłączony. Zachowanie właściwości: - *wymagane*

templatedashboard.create

Opis

`object templatedashboard.create(object/array templateDashboards)`

Ta metoda umożliwia tworzenie nowych pulpitów nawigacyjnych szablonu.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Pulpity szablonu do utworzenia.

Oprócz standardowych właściwości pulpitu szablonu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
pages	array	Strony pulpitu szablonu, które mają zostać utworzone dla pulpitu. Strony pulpitu będą uporządkowane w tej samej kolejności, w jakiej zostały podane. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych pulpitów nawigacyjnych szablonu we właściwości `dashboardids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych pulpitów nawigacyjnych szablonu.

Przykłady

Tworzenie dashboardu szablonu

Utwórz dashboard szablonu o nazwie „Graphs” z jednym widżetem Graph na pojedynczej stronie dashboardu.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "32"
        ]
    },
    "id": 1
}
```

Zobacz także

- Strona pulpitu szablonu
- Widżet pulpitu szablonu
- Pole widżetu pulpitu szablonu

Źródło

CTemplateDashboard::create() w *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.delete

Opis

`object templatedashboard.delete(array templateDashboardIds)`

Ta metoda umożliwia usuwanie pulpitów nawigacyjnych szablonu.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID pulpitów szablonu do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych pulpitów szablonu we właściwości `dashboardids`.

Przykłady

Usuwanie wielu pulpitów szablonów

Usuń dwa pulpity szablonów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "templateashboard.delete",
"params": [
"45",
"46"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"dashboardids": [
"45",
"46"
]
},
"id": 1
}
```

Źródło

CTemplateDashboard::delete() w *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.get

Opis

`integer/array templatedashboard.get(object parameters)`

Metoda umożliwia pobieranie pulpitów nawigacyjnych szablonów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dashboardids	ID/array	Zwraca tylko dashboardy szablonów o podanych identyfikatorach.
templateids	ID/array	Zwraca tylko dashboardy szablonów należące do podanych szablonów.
selectPages	query	Zwraca właściwość `pages` ze stronami dashboardu szablonu, w poprawnej kolejności.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dashboardid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie pulpitów szablonu

Pobierz wszystkie pulpity szablonu z widgetami dla określonego szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.get",
    "params": {
        "output": "extend",
        "selectPages": "extend",
        "templateids": "10001"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Strona pulpitu szablonu
- Widżet pulpitu szablonu
- Pole widżetu pulpitu szablonu

Źródło

CTemplateDashboard::get() w *ui/include/classes/api/services/CTemplateDashboard.php*.

templatedashboard.update

Opis

`object templatedashboard.update(object/array templateDashboards)`

Ta metoda umożliwia aktualizację istniejących pulpitów nawigacyjnych szablonu.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości pulpitu nawigacyjnego szablonu do zaktualizowania.

Właściwość dashboardid musi być określona dla każdego pulpitu nawigacyjnego, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko określone właściwości.

Oprócz standardowych właściwości pulpitu nawigacyjnego szablonu, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
pages	array	Strony pulpitu nawigacyjnego szablonu, które zastąpią istniejące strony pulpitu nawigacyjnego. Strony pulpitu nawigacyjnego są aktualizowane za pomocą właściwości dashboard_pageid. Nowe strony pulpitu nawigacyjnego zostaną utworzone dla obiektów bez właściwości dashboard_pageid, a istniejące strony pulpitu nawigacyjnego zostaną usunięte, jeśli nie zostaną ponownie użyte. Strony pulpitu nawigacyjnego zostaną uporządkowane w tej samej kolejności, w jakiej zostały określone. Zaktualizowane zostaną tylko określone właściwości stron pulpitu nawigacyjnego. Dla właściwości pages wymagany jest co najmniej jeden obiekt strony pulpitu nawigacyjnego.

Wartości zwracane

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych pulpitów szablonu we właściwości dashboardids.

Przykłady

Zmiana nazwy dashboardu szablonu

Zmień nazwę dashboardu szablonu na „Performance graphs”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.update",
    "params": {
        "dashboardid": "23",
        "name": "Performance graphs"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "23"
        ]
    },
    "id": 1
}
```

Aktualizowanie stron pulpitu nawigacyjnego szablonu

Zmień nazwę pierwszej strony pulpitu nawigacyjnego, zastąp widgety na drugiej stronie pulpitu nawigacyjnego i dodaj nową stronę jako trzecią. Usuń wszystkie pozostałe strony pulpitu nawigacyjnego.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "templatedashboard.update",
    "params": {
        "dashboardid": "2",
        "pages": [
            {
                "dashboard_pageid": 1,
```

```
                "name": "Renamed Page"
            },
            {
                "dashboard_pageid": 2,
                "widgets": [
                    {
                        "type": "clock",
                        "x": 0,
                        "y": 0,
                        "width": 12,
                        "height": 3
                    }
                ]
            },
            {
                "display_period": 60
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}
```

Zobacz także

- Widżet pulpitu szablonu
- Pole widżetu pulpitu szablonu

Źródło

CTemplateDashboard::update() w *ui/include/classes/api/services/CTemplateDashboard.php*.

Raport

Ta klasa jest przeznaczona do pracy z zaplanowanymi raportami.

Odwołania do obiektów:

- Report
- Users
- User groups

Dostępne metody:

- report.create - tworzenie nowych zaplanowanych raportów
- report.delete - usuwanie zaplanowanych raportów
- report.get - pobieranie zaplanowanych raportów
- report.update - aktualizowanie zaplanowanych raportów

Obiekt raportu

Następujące obiekty są bezpośrednio powiązane z API `report`.

Raport

Obiekt raportu ma następujące właściwości:

Property	Type	Description
reportid	ID	Identyfikator raportu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
userid	ID	Identyfikator użytkownika, który utworzył raport. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
name	string	Unikalna nazwa raportu. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
dashboardid	ID	Identyfikator dashboardu, na podstawie którego tworzony jest raport. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
period	integer	Okres, dla którego zostanie przygotowany raport. Możliwe wartości: 0 - *(domyślnie)* poprzedni dzień; 1 - poprzedni tydzień; 2 - poprzedni miesiąc; 3 - poprzedni rok.
cycle	integer	Okres powtarzania harmonogramu. Możliwe wartości: 0 - *(domyślnie)* codziennie; 1 - co tydzień; 2 - co miesiąc; 3 - co rok.
start_time	integer	Pora dnia, w sekundach, o której raport zostanie przygotowany do wysłania. Wartość domyślna: 0.
weekdays	integer	Dni tygodnia, w których raport będzie wysyłany. Możliwe wartości bitmapowe: 1 - poniedziałek; 2 - wtorek; 4 - środa; 8 - czwartek; 16 - piątek; 32 - sobota; 64 - niedziela. To pole jest maską bitową; dopuszczalna jest dowolna suma możliwych wartości bitmapowych (na przykład 21 oznacza poniedziałek, środę i piątek). Wartość domyślna: 0.
active_since	string	Data rozpoczęcia. Zachowanie właściwości: - *wymagane*, jeśli `cycle` ma wartość „weekly”. Możliwe wartości: pusty ciąg - *(domyślnie)* nie określono (przechowywane jako 0); konkretna data w formacie YYYY-MM-DD (przechowywana jako znacznik czasu początku dnia (00:00:00)).

Property	Type	Description
active_till	string	Data zakończenia. Możliwe wartości: pusty ciąg - *(domyślnie)* nie określono (przechowywane jako 0); konkretna data w formacie YYYY-MM-DD (przechowywana jako znacznik czasu końca dnia (23:59:59)).
subject	string	Temat wiadomości raportu.
message	string	Treść wiadomości raportu.
status	integer	Określa, czy raport jest włączony, czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
description	text	Opis raportu.
state	integer	Stan raportu. Możliwe wartości: 0 - *(domyślnie)* raport nie został jeszcze przetworzony; 1 - raport został wygenerowany i pomyślnie wysłany do wszystkich odbiorców; 2 - generowanie raportu nie powiodło się; właściwość „info” zawiera informacje o błędzie; 3 - raport został wygenerowany, ale wysyłanie do niektórych lub wszystkich odbiorców nie powiodło się; właściwość „info” zawiera informacje o błędzie. Zachowanie właściwości: - *tylko do odczytu*
lastsent	timestamp	Znacznik czasu Unix ostatniego pomyślnie wysłanego raportu. Zachowanie właściwości: - *tylko do odczytu*
info	string	Opis błędu lub dodatkowe informacje. Zachowanie właściwości: - *tylko do odczytu*

Użytkownicy

Obiekt users ma następujące właściwości:

Właściwość	Typ	Opis
userid	ID	ID użytkownika, do którego ma zostać wysłany raport. Zachowanie właściwości: - *wymagane*
access_userid	ID	ID użytkownika, w imieniu którego raport zostanie wygenerowany. 0 - *(domyślnie)* Wygeneruj raport przez odbiorcę.
exclude	integer	Czy wykluczyć użytkownika z listy mailingowej. Możliwe wartości: 0 - *(domyślnie)* Uwzględnij; 1 - Wyklucz.

Grupy użytkowników

Obiekt grup użytkowników ma następujące właściwości:

Właściwość	Typ	Opis
usrgrpid	ID	ID grupy użytkowników, do której ma zostać wysłany raport. Zachowanie właściwości: - *wymagane*
access_userid	ID	ID użytkownika, w imieniu którego zostanie wygenerowany raport. 0 - *(domyślnie)* Wygeneruj raport przez odbiorcę.

report.create

Opis

`object report.create(object/array reports)`

Ta metoda umożliwia tworzenie nowych zaplanowanych raportów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Zaplanowane raporty do utworzenia.

Oprócz standardowych właściwości zaplanowanego raportu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
users	object/array	Użytkownicy, do których ma zostać wysłany raport. Zachowanie parametrów: - *wymagane*, jeśli `user_groups` nie jest ustawione
user_groups	object/array	Grupy użytkowników, do których ma zostać wysłany raport. Zachowanie parametrów: - *wymagane*, jeśli `users` nie jest ustawione

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych zaplanowanych raportów we właściwości `reportids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych zaplanowanych raportów.

Przykłady

Tworzenie zaplanowanego raportu

Utwórz raport tygodniowy, który będzie przygotowywany za poprzedni tydzień w każdy poniedziałek-piątek o 12:00 od 2021-04-01 do 2021-08-31.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.create",
    "params": {
        "userid": "1",
        "name": "Raport tygodniowy",
        "dashboardid": "1",
        "period": "1",
        "cycle": "1",
        "start_time": "43200",
        "weekdays": "31",
        "active_since": "2021-04-01",
        "active_till": "2021-08-31",
```

```
        "subject": "Raport tygodniowy",
        "message": "Tekst towarzyszący raportowi",
        "status": "1",
        "description": "Opis raportu",
        "users": [
            {
                "userid": "1",
                "access_userid": "1",
                "exclude": "0"
            },
            {
                "userid": "2",
                "access_userid": "0",
                "exclude": "1"
            }
        ],
        "user_groups": [
            {
                "usrgrpid": "7",
                "access_userid": "0"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Użytkownicy
- Grupy użytkowników

Źródło

CReport::create() w *ui/include/classes/api/services/CReport.php*.

report.delete

Opis

`object report.delete(array reportids)`

Ta metoda umożliwia usuwanie zaplanowanych raportów.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID raportów okresowych do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych zaplanowanych raportów we właściwości `reportids`.

Przykłady

Usuwanie wielu zaplanowanych raportów

Usuń dwa zaplanowane raporty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.delete",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1",
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CReport::delete() w *ui/include/classes/api/services/CReport.php*.

report.get

Opis

`integer/array report.get(object parameters)`

Metoda umożliwia pobieranie zaplanowanych raportów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
reportids	ID/array	Zwraca tylko zaplanowane raporty o podanych identyfikatorach raportów.
expired	boolean	Jeśli ustawiono na `true`, zwraca tylko wygasłe zaplanowane raporty.
selectUsers	query	Zwraca właściwość `users` zawierającą użytkowników, do których skonfigurowano wysyłanie raportu.
selectUserGroups	query	Zwraca właściwość `user_groups` zawierającą grupy użytkowników, do których skonfigurowano wysyłanie raportu.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `reportid`, `name`, `status`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	

Parametr	Typ	Opis
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych raportów

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.get",
    "params": {
        "output": "extend",
        "selectUsers": "extend",
        "selectUserGroups": "extend",
        "reportids": ["1", "2"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "reportid": "1",
            "userid": "1",
            "name": "Raport tygodniowy",
            "dashboardid": "1",
            "period": "1",
            "cycle": "1",
            "start_time": "43200",
            "weekdays": "31",
            "active_since": "2021-04-01",
            "active_till": "2021-08-31",
            "subject": "Raport tygodniowy",
            "message": "Tekst towarzyszący raportowi",
            "status": "1",
            "description": "Opis raportu",
            "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": "Raport miesięczny",
            "dashboardid": "2",
            "period": "2",
            "cycle": "2",
            "start_time": "0",
            "weekdays": "0",
            "active_since": "2021-05-01",
            "active_till": "",
            "subject": "Raport miesięczny",
            "message": "Tekst towarzyszący raportowi",
            "status": "1",
            "description": "",
            "state": "0",
            "lastsent": "0",
            "info": "",
            "users": [
                {
                    "userid": "1",
                    "access_userid": "1",
                    "exclude": "0"
                }
            ],
            "user_groups": []
        }
    ],
    "id": 1
}
```


Zobacz także

- Użytkownicy
- Grupy użytkowników

Źródło

CReport::get() w *ui/include/classes/api/services/CReport.php*.

report.update

Opis

`object report.update(object/array reports)`

Ta metoda umożliwia aktualizację istniejących zaplanowanych raportów.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości zaplanowanego raportu do zaktualizowania.

Właściwość `reportid` musi być zdefiniowana dla każdego zaplanowanego raportu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości zaplanowanego raportu metoda akceptuje następujące parametry.

Parametr	Typ	Opis
users	object/array	Użytkownicy, którzy mają zastąpić bieżących użytkowników przypisanych do zaplanowanego raportu. Zachowanie parametru: - *wymagany*, jeśli `user_groups` nie jest ustawione
user_groups	object/array	Grupy użytkowników, które mają zastąpić bieżące grupy użytkowników przypisane do zaplanowanego raportu. Zachowanie parametru: - *wymagany*, jeśli `users` nie jest ustawione

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych zaplanowanych raportów we właściwości `reportids`.

Przykłady

Wyłączanie zaplanowanego raportu

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "report.update",
    "params": {
        "reportid": "1",
        "status": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}
```

Zobacz także

- Użytkownicy
- Grupy użytkowników

Źródło

CReport::update() w *ui/include/classes/api/services/CReport.php*.

Reguła LLD

Ta klasa jest przeznaczona do pracy z regułami wykrywania niskiego poziomu.

Odwołania do obiektów:

- Reguła LLD

    - Nagłówek HTTP
    - Pole zapytania HTTP
- Filtr reguły LLD
    - Warunek filtra reguły LLD
- Ścieżka makra LLD
- Przetwarzanie wstępne reguły LLD
- Nadpisania reguły LLD
    - Filtr nadpisania reguły LLD
        * Warunek filtra nadpisania reguły LLD
    - Operacja nadpisania reguły LLD
        * Status operacji nadpisania reguły LLD
        * Wykrywanie operacji nadpisania reguły LLD
        * Okres operacji nadpisania reguły LLD
        * Historia operacji nadpisania reguły LLD
        * Trendy operacji nadpisania reguły LLD
        * Poziom ważności operacji nadpisania reguły LLD
        * Tag operacji nadpisania reguły LLD
        * Szablon operacji nadpisania reguły LLD
        * Inwentaryzacja operacji nadpisania reguły LLD

Dostępne metody:

- discoveryrule.create - utwórz nowe reguły LLD
- discoveryrule.delete - usuń reguły LLD
- discoveryrule.get - pobierz reguły LLD
- discoveryrule.update - zaktualizuj reguły LLD

Obiekt reguły LLD (Low Level Discovery)

Następujące obiekty są bezpośrednio powiązane z API `discoveryrule`.

Reguła LLD

Obiekt reguły wykrywania niskiego poziomu ma następujące właściwości.

Właściwość	Typ	Opis
itemid	ID	ID reguły LLD. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* dla operacji aktualizacji
delay	string	Interwał aktualizacji reguły LLD. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub opcjonalnie jeden lub więcej niestandardowych interwałów, rozdzielonych średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika. Jeśli jest używane, wartość musi być pojedynczym makrem. Wiele makr ani makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisać jako dwa makra rozdzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „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) lub „Browser” (22), albo jeśli `type` ma wartość „Zabbix agent (active)” (7), a `key_` nie zawiera „mqtt.get”

Właściwość	Typ	Opis
hostid	ID	ID hosta, do którego należy reguła LLD. Zachowanie właściwości: - *stała* - *wymagana* dla operacji tworzenia
flags	integer	Pochodzenie reguły wykrywania. Możliwe wartości: 1 - reguła wykrywania niskiego poziomu; 5 - reguła wykrywania niskiego poziomu przekonwertowana z prototypu. Zachowanie właściwości: - *tylko do odczytu*
interfaceid	ID	ID interfejsu hosta reguły LLD. Zachowanie właściwości: - *wymagana*, jeśli reguła LLD należy do hosta, a `type` ma wartość „Zabbix agent”, „IPMI agent”, „JMX agent” lub „SNMP agent” - *obsługiwana*, jeśli reguła LLD należy do hosta, a `type` ma wartość „Simple check”, „External check”, „SSH agent”, „TELNET agent” lub „HTTP agent”
key_	string	Klucz reguły LLD. Zachowanie właściwości: - *wymagana* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
name	string	Nazwa reguły LLD. Zachowanie właściwości: - *wymagana* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych
type	integer	Typ reguły LLD. Możliwe wartości: 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. Ten typ jest dozwolony tylko wtedy, gdy element nadrzędny reguły jest szablonem (zakłada się, że podczas wykrywania jest połączony z prototypem hosta) lub wykrytym hostem (który ma regułę LLD w katalogu głównym). Zachowanie właściwości: - *wymagana* dla operacji tworzenia - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
url	string	Ciąg URL. Obsługuje makra użytkownika, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {ITEM.ID}, {ITEM.KEY}. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
allow_traps	integer	Zezwala na wypełnianie wartości w sposób podobny do pozycji trappera. Możliwe wartości: 0 - *(domyślnie)* nie akceptuj danych przychodzących; 1 - akceptuj dane przychodzące. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent”
authtype	integer	Metoda uwierzytelniania. Możliwe wartości, jeśli `type` ma wartość „SSH agent”: 0 - *(domyślnie)* hasło; 1 - klucz publiczny. Możliwe wartości, jeśli `type` ma wartość „HTTP agent”: 0 - *(domyślnie)* brak; 1 - podstawowe; 2 - NTLM; 3 - Kerberos; 4 - Digest. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „SSH agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
description	string	Opis reguły LLD.
error	string	Tekst błędu, jeśli występują problemy z aktualizacją wartości reguły LLD. Zachowanie właściwości: - *tylko do odczytu*
follow_redirects	integer	Śledzenie przekierowań odpowiedzi podczas odpytywania danych. Możliwe wartości: 0 - nie śledź przekierowań; 1 - *(domyślnie)* śledź przekierowania. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
headers	array	Tablica nagłówków, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
http_proxy	string	Ciąg połączenia proxy HTTP(S). Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
ipmi_sensor	string	Czujnik IPMI. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „IPMI agent”, a `key_` nie ma wartości „ipmi.get” - *obsługiwana*, jeśli `type` ma wartość „IPMI agent” - *tylko do odczytu* dla obiektów dziedziczonych
jmx_endpoint	string	Niestandardowy ciąg połączenia agenta JMX. Wartość domyślna: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „JMX agent”
lifetime	string	Okres, po którym pozycje, które nie są już wykrywane, zostaną usunięte. Akceptuje liczbę sekund (np. 3600), wartość z przyrostkiem czasu (np. `3600s`, `180m`, `24h`, `1d`) lub makro użytkownika. Wartość domyślna: `7d`.
lifetime_type	integer	Scenariusz usuwania utraconych zasobów LLD. Możliwe wartości: 0 - *(domyślnie)* usuń po osiągnięciu progu okresu przechowywania; 1 - nie usuwaj; 2 - usuń natychmiast.
enabled_lifetime	string	Okres, po którym pozycje, które nie są już wykrywane, zostaną wyłączone. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`) lub makro użytkownika. Wartość domyślna: 0.
enabled_lifetime_type	integer	Scenariusz wyłączania utraconych zasobów LLD. Możliwe wartości: 0 - wyłącz po osiągnięciu progu okresu przechowywania; 1 - nie wyłączaj; 2 - *(domyślnie)* wyłącz natychmiast.
master_itemid	ID	ID pozycji nadrzędnej. Reguła wykrywania nie może być pozycją nadrzędną dla innej reguły wykrywania. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Dependent item” - *tylko do odczytu* dla obiektów dziedziczonych
output_format	integer	Czy odpowiedź powinna zostać przekonwertowana do formatu JSON. Możliwe wartości: 0 - *(domyślnie)* przechowuj dane surowe; 1 - konwertuj do formatu JSON. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
params	string	Dodatkowe parametry zależne od typu reguły LLD: - wykonywany skrypt dla reguł LLD SSH i Telnet; - zapytanie SQL dla reguł LLD monitora bazy danych; - formuła dla obliczanych reguł LLD; - skrypt dla reguł LLD skryptu i przeglądarki. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Database monitor”, „SSH agent”, „TELNET agent”, „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „Script” lub „Browser”)
parameters	object/array	Dodatkowe parametry, jeśli `type` ma wartość „Script” lub „Browser”. Tablica obiektów z właściwościami `name` i `value`, gdzie `name` musi być unikatowe. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Script” lub „Browser” - *tylko do odczytu* dla obiektów dziedziczonych
password	string	Hasło do uwierzytelniania. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „JMX agent”, a `username` jest ustawione - *obsługiwana*, jeśli `type` ma wartość „Simple check”, „Database monitor”, „SSH agent”, „TELNET agent” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
post_type	integer	Typ treści danych POST przechowywanej we właściwości `posts`. Możliwe wartości: 0 - *(domyślnie)* dane surowe; 2 - dane JSON; 3 - dane XML. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
posts	string	Dane treści żądania HTTP(S). Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „JSON data” lub „XML data” - *obsługiwana*, jeśli `type` ma wartość „HTTP agent”, a `post_type` ma wartość „Raw data” - *tylko do odczytu* dla obiektów dziedziczonych
privatekey	string	Nazwa pliku klucza prywatnego. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key”
publickey	string	Nazwa pliku klucza publicznego. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „SSH agent”, a `authtype` ma wartość „public key”
query_fields	array	Tablica pól zapytania, które zostaną wysłane podczas wykonywania żądania HTTP. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Właściwość	Typ	Opis
request_method	integer	Typ metody żądania. Możliwe wartości: 0 - *(domyślnie)* GET; 1 - POST; 2 - PUT; 3 - HEAD. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
retrieve_mode	integer	Część odpowiedzi, która powinna zostać zapisana. Możliwe wartości, jeśli `request_method` ma wartość „GET”, „POST” lub „PUT”: 0 - *(domyślnie)* treść; 1 - nagłówki; 2 - zostaną zapisane zarówno treść, jak i nagłówki. Możliwe wartości, jeśli `request_method` ma wartość „HEAD”: 1 - nagłówki. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
snmp_oid	string	OID SNMP. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „SNMP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_cert_file	string	Ścieżka do pliku publicznego klucza SSL. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_file	string	Ścieżka do pliku prywatnego klucza SSL. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
ssl_key_password	string	Hasło do pliku klucza SSL. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
state	integer	Stan reguły LLD. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - nieobsługiwana. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Stan reguły LLD. Możliwe wartości: 0 - *(domyślnie)* włączona reguła LLD; 1 - wyłączona reguła LLD.

Właściwość	Typ	Opis
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielone przecinkami. Obsługuje również makra użytkownika jako część listy rozdzielanej przecinkami. Przykład: 200,200-{$M},{$M},200-400 Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
templateid	ID	ID nadrzędnej reguły LLD szablonu. Zachowanie właściwości: - *tylko do odczytu*
timeout	string	Limit czasu żądania odpytywania danych pozycji. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m). Akceptuje również makra użytkownika. Zakres możliwych wartości: 1-600s. Wartość domyślna: ”” - użyj ustawień proxy/globalnych. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Zabbix agent” (0), „Simple check” (3), a `key_` nie zaczyna się od „`vmware.`” ani „`icmpping`”, „Zabbix agent (active)” (7), „External check” (10), „Database monitor” (11), „SSH agent” (13), „TELNET agent” (14), „HTTP agent” (19), „SNMP agent” (20), a `snmp_oid` zaczyna się od „`walk[`” lub „`get[`”, „Script” (21) albo „Browser” (22) - *tylko do odczytu* dla obiektów dziedziczonych, - jeśli zastosowano wartość domyślną, a pozycja należy do hosta monitorowanego przez niedostępny proxy lub grupę proxy, używany jest globalny limit czasu.
trapper_hosts	string	Dozwolone hosty. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Zabbix trapper” lub jeśli `type` ma wartość „HTTP agent”, a `allow_traps` ma wartość „Allow incoming data”. Jeśli właściwość zostanie pominięta, zostanie automatycznie ustawiona na `{$TRAPPER.ALLOWED_HOSTS}` makro użytkownika (wartość domyślna). Domyślnie `{$TRAPPER.ALLOWED_HOSTS}` ma wartość `127.0.0.1,::1`. Jeśli właściwość zostanie ustawiona na pusty ciąg, połączenia przychodzące ze wszystkich hostów będą odrzucane.
username	string	Nazwa użytkownika do uwierzytelniania. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „SSH agent”, „TELNET agent” lub jeśli `type` ma wartość „JMX agent”, a `password` jest ustawione - *obsługiwana*, jeśli `type` ma wartość „Simple check”, „Database monitor” lub „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych (jeśli `type` ma wartość „HTTP agent”)
uuid	string	Uniwersalny unikatowy identyfikator używany do łączenia zaimportowanych reguł LLD z już istniejącymi regułami. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwana*, jeśli reguła LLD należy do szablonu

Właściwość	Typ	Opis
verify_host	integer	Czy należy sprawdzić, czy nazwa hosta używana dla połączenia jest zgodna z nazwą w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* nie sprawdzaj; 1 - sprawdzaj. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych
verify_peer	integer	Czy należy sprawdzić, czy certyfikat hosta jest autentyczny. Możliwe wartości: 0 - *(domyślnie)* nie sprawdzaj; 1 - sprawdzaj. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „HTTP agent” - *tylko do odczytu* dla obiektów dziedziczonych

Nagłówek HTTP

Obiekt nagłówka ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa nagłówka HTTP. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka. Zachowanie właściwości: - *wymagane*

Pole zapytania HTTP

Obiekt pola zapytania definiuje nazwę i wartość używane do określenia parametru URL. Ma następujące właściwości:

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Zachowanie właściwości: - *wymagane*

Filtr reguły LLD

Obiekt filtra reguły LLD definiuje zestaw warunków, które mogą być używane do filtrowania wykrytych obiektów. Ma następujące właściwości:

Właściwość	Typ	Opis
conditions	object/array	Zestaw warunków filtra używanych do filtrowania wyników. Warunki zostaną posortowane zgodnie z kolejnością ich występowania w formule. Zachowanie właściwości: - *wymagane*
evaltype	integer	Metoda oceny warunków filtra. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra. Wyrażenie zawiera identyfikatory odnoszące się do określonych warunków filtra przez ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odnoszące się do określonych warunków filtra przez ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „custom expression”

Warunek filtra reguły LLD

Obiekt warunku filtra reguły LLD definiuje oddzielne sprawdzenie wykonywane na wartości makra LLD. Ma następujące właściwości:

Właściwość	Type	Opis
macro	string	Makro LLD, na którym ma zostać wykonane sprawdzenie. Property behavior: - *required*
value	string	Wartość do porównania. Property behavior: - *required* jeśli `operator` jest ustawiony na „matches regular expression” lub „does not match regular expression”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym ich pobieraniu. Property behavior: - *required* jeśli `evaltype` obiektu LLD rule filter object jest ustawiony na „custom expression”

Właściwość	Type	Opis
operator	integer	Operator warunku. Możliwe wartości: 8 - *(domyślnie)* pasuje do wyrażenia regularnego; 9 - nie pasuje do wyrażenia regularnego; 12 - istnieje; 13 - nie istnieje.

Note:
Aby lepiej zrozumieć, jak używać filtrów z różnymi typami wyrażeń, zobacz przykłady na stronach metod discoveryrule.get i discoveryrule.create.

Ścieżka makra LLD

Ścieżka makra LLD ma następujące właściwości:

Właściwość	Type	Opis
lld_macro	string	Makro LLD. Zachowanie właściwości: - *wymagane*
path	string	Selektor wartości, która zostanie przypisana do odpowiadającego jej makra. Zachowanie właściwości: - *wymagane*

Wstępne przetwarzanie reguły LLD

Obiekt wstępnego przetwarzania reguły LLD ma następujące właściwości.

Property	Type	Description
type	integer	Typ opcji wstępnego przetwarzania. Możliwe wartości: 5 - Wyrażenie regularne; 11 - XML XPath; 12 - JSONPath; 14 - Pasuje do wyrażenia regularnego; 15 - Nie pasuje do wyrażenia regularnego; 16 - Sprawdzenie błędu w JSON; 17 - Sprawdzenie błędu w XML; 20 - Odrzucenie niezmienionych wartości z sygnałem heartbeat; 21 - JavaScript; 23 - Prometheus do JSON; 24 - CSV do JSON; 25 - Zastąpienie; 27 - XML do JSON; 28 - Wartość SNMP walk; 29 - SNMP walk do JSON; 30 - Pobranie wartości SNMP. Property behavior: - *required*

Property	Type	Description
params	string	Dodatkowe parametry używane przez opcję wstępnego przetwarzania. Wiele parametrów jest rozdzielanych znakiem nowego wiersza (\n). Property behavior: - *required* jeśli `type` ma wartość „Wyrażenie regularne” (5), „XML XPath” (11), „JSONPath” (12), „Pasuje do wyrażenia regularnego” (14), „Nie pasuje do wyrażenia regularnego” (15), „Sprawdzenie błędu w JSON” (16), „Sprawdzenie błędu w XML” (17), „Odrzucenie niezmienionych wartości z sygnałem heartbeat” (20), „JavaScript” (21), „Prometheus do JSON” (23), „CSV do JSON” (24), „Zastąpienie” (25), „Wartość SNMP walk” (28), „SNMP walk do JSON” (29) lub „Pobranie wartości SNMP” (30)
error_handler	integer	Typ działania używanego w przypadku niepowodzenia kroku wstępnego przetwarzania. Możliwe wartości: 0 - Komunikat o błędzie jest ustawiany przez serwer Zabbix; 1 - Odrzucenie wartości; 2 - Ustawienie wartości niestandardowej; 3 - Ustawienie niestandardowego komunikatu o błędzie. Property behavior: - *required* jeśli `type` ma wartość „Wyrażenie regularne” (5), „XML XPath” (11), „JSONPath” (12), „Pasuje do wyrażenia regularnego” (14), „Nie pasuje do wyrażenia regularnego” (15), „Sprawdzenie błędu w JSON” (16), „Sprawdzenie błędu w XML” (17), „Prometheus do JSON” (23), „CSV do JSON” (24), „XML do JSON” (27), „Wartość SNMP walk” (28), „SNMP walk do JSON” (29) lub „Pobranie wartości SNMP” (30)
error_handler_params	string	Parametry obsługi błędów. Property behavior: - *required* jeśli `error_handler` ma wartość „Ustawienie wartości niestandardowej” lub „Ustawienie niestandardowego komunikatu o błędzie”

Dla każdego typu wstępnego przetwarzania obsługiwane są następujące parametry i procedury obsługi błędów.

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
5	Wyrażenie regularne	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	Pasuje do wyrażenia regularnego	pattern[1]			0, 1, 2, 3
15	Nie pasuje do wyrażenia regularnego	pattern[1]			0, 1, 2, 3

Preprocessing type	Name	Parameter 1	Parameter 2	Parameter 3	Supported error handlers
16	Sprawdzenie błędu w JSON	path[3]			0, 1, 2, 3
17	Sprawdzenie błędu w XML	path[3]			0, 1, 2, 3
20	Odrzucenie niezmienionych wartości z sygnałem heartbeat	seconds^4, 5^			
21	JavaScript	script[2]			
23	Prometheus do JSON	pattern^5, 6^			0, 1, 2, 3
24	CSV do JSON	character[2]	character[2]	0,1	0, 1, 2, 3
25	Zastąpienie	search string[2]	replacement[2]		
27	XML do JSON				0, 1, 2, 3
28	Wartość OID SNMP walk	OID[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS		0, 1, 2, 3
29	SNMP walk do JSON[7]	Field name[2]	OID prefix[2]	Format: 0 - Bez zmian 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS	0, 1, 2, 3
30	Pobranie wartości SNMP	Format: 1 - UTF-8 z Hex-STRING 2 - MAC z Hex-STRING 3 - Liczba całkowita z BITS			0, 1, 2, 3

[1] wyrażenie regularne
 [2] ciąg znaków
 [3] JSONPath lub XML XPath
 [4] dodatnia liczba całkowita (z obsługą sufiksów czasu, np. 30s, 1m, 2h, 1d)
 [5] makro użytkownika
 [6] wzorzec Prometheus zgodny ze składnią: `<metric name>{<label name>="<label value>", ...} == <value>`. Każdy składnik wzorca Prometheus (metryka, nazwa etykiety, wartość etykiety i wartość metryki) może być makrem użytkownika.
 [7] Obsługuje wiele rekordów „Field name,OID prefix,Format” rozdzielonych znakiem nowego wiersza.

Nadpisania reguł LLD

Obiekt nadpisań reguł LLD definiuje zestaw reguł (filtrów, warunków i operacji), które są używane do nadpisywania właściwości różnych obiektów prototypów. Ma następujące właściwości:

Właściwość	Type	Opis
name	string	Unikalna nazwa nadpisania. Zachowanie właściwości: - *wymagane*
step	integer	Unikalny numer kolejności nadpisania. Zachowanie właściwości: - *wymagane*
stop	integer	Zatrzymaj przetwarzanie kolejnych nadpisań, jeśli wystąpi dopasowanie. Możliwe wartości: 0 - *(domyślnie)* nie zatrzymuj przetwarzania nadpisań; 1 - zatrzymaj przetwarzanie nadpisań, jeśli filtr pasuje.
filter	object	Filtr nadpisania.
operations	object/array	Operacje nadpisania.

Filtr nadpisania reguły LLD

Obiekt filtra nadpisania reguły LLD definiuje zestaw warunków, które po dopasowaniu do wykrytego obiektu powodują zastosowanie nadpisania. Ma następujące właściwości:

Właściwość	Typ	Opis
conditions	object/array	Zestaw warunków filtra nadpisania używanych do dopasowywania wykrytych obiektów. Warunki zostaną posortowane zgodnie z kolejnością ich umieszczenia w formule. Zachowanie właściwości: - *wymagane*
evaltype	integer	Metoda oceny warunków filtra nadpisania. Możliwe wartości: 0 - And/Or; 1 - And; 2 - Or; 3 - Wyrażenie niestandardowe. Zachowanie właściwości: - *wymagane*
eval_formula	string	Wygenerowane wyrażenie, które będzie używane do oceny warunków filtra nadpisania. Wyrażenie zawiera identyfikatory odwołujące się do określonych warunków filtra nadpisania za pomocą ich `formulaid`. Wartość `eval_formula` jest równa wartości `formula` dla filtrów z wyrażeniem niestandardowym. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
formula	string	Wyrażenie zdefiniowane przez użytkownika, używane do oceny warunków filtrów nadpisania z wyrażeniem niestandardowym. Wyrażenie musi zawierać identyfikatory odwołujące się do określonych warunków filtra nadpisania za pomocą ich `formulaid`. Identyfikatory użyte w wyrażeniu muszą dokładnie odpowiadać tym zdefiniowanym w warunkach filtra nadpisania: żaden warunek nie może pozostać nieużyty ani pominięty. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` jest ustawione na „wyrażenie niestandardowe”

Warunek filtra nadpisania reguły LLD

Obiekt warunku filtra nadpisania reguły LLD definiuje oddzielne sprawdzenie wykonywane na wartości makra LLD. Ma następujące właściwości:

Właściwość	Type	Opis
macro	string	Makro LLD, dla którego ma zostać wykonane sprawdzenie. Zachowanie właściwości: - *wymagane*
value	string	Wartość do porównania. Zachowanie właściwości: - *wymagane*, jeśli `operator` jest ustawiony na „pasuje do wyrażenia regularnego” lub „nie pasuje do wyrażenia regularnego”
formulaid	string	Dowolny unikalny identyfikator używany do odwoływania się do warunku z niestandardowego wyrażenia. Może zawierać wyłącznie wielkie litery. Identyfikator musi zostać zdefiniowany przez użytkownika podczas modyfikowania warunków filtra, ale zostanie wygenerowany ponownie przy późniejszym żądaniu tych warunków. Zachowanie właściwości: - *wymagane*, jeśli `evaltype` obiektu filtra nadpisania reguły LLD jest ustawiony na „wyrażenie niestandardowe”
operator	integer	Operator warunku. Możliwe wartości: 8 - *(domyślnie)* pasuje do wyrażenia regularnego; 9 - nie pasuje do wyrażenia regularnego; 12 - istnieje; 13 - nie istnieje.

Operacja nadpisania reguły LLD

Operacja nadpisania reguły LLD to połączenie warunków i działań wykonywanych na obiekcie prototypu. Ma następujące właściwości:

Właściwość	Type	Opis
operationobject	integer	Typ wykrytego obiektu, dla którego ma zostać wykonane działanie. Możliwe wartości: 0 - Prototyp pozycji; 1 - Prototyp wyzwalacza; 2 - Prototyp wykresu; 3 - Prototyp hosta; 4 - Prototyp wykrywania. Zachowanie właściwości: - *wymagane*
operator	integer	Operator warunku nadpisania. Możliwe wartości: 0 - *(domyślnie)* równa się; 1 - nie równa się; 2 - zawiera; 3 - nie zawiera; 8 - pasuje; 9 - nie pasuje.
value	string	Wzorzec do dopasowania do nazwy prototypu pozycji, wyzwalacza, wykresu lub hosta, zależnie od wybranego obiektu.
opstatus	object	Obiekt statusu operacji nadpisania dla obiektów prototypu pozycji, wyzwalacza i hosta.
opdiscover	object	Obiekt statusu wykrywania operacji nadpisania (wszystkie typy obiektów).
opperiod	object	Obiekt okresu operacji nadpisania (interwału aktualizacji) dla obiektu prototypu pozycji.
ophistory	object	Obiekt historii operacji nadpisania dla obiektu prototypu pozycji.
optrends	object	Obiekt trendów operacji nadpisania dla obiektu prototypu pozycji.
opseverity	object	Obiekt ważności operacji nadpisania dla obiektu prototypu wyzwalacza.
optag	object/array	Obiekt tagu operacji nadpisania dla obiektów prototypu wyzwalacza i hosta.
optemplate	object/array	Obiekt szablonu operacji nadpisania dla obiektu prototypu hosta.
opinventory	object	Obiekt inwentaryzacji operacji nadpisania dla obiektu prototypu hosta.

Status operacji nadpisania reguły LLD

Status operacji nadpisania reguły LLD ustawiany dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Typ	Opis
status	integer	Nadpisuje status wybranego obiektu. Możliwe wartości: 0 - Utwórz włączony; 1 - Utwórz wyłączony. Zachowanie właściwości: - *wymagane*

Operacja nadpisania reguły LLD: discover

Status discover operacji nadpisania reguły LLD, który jest ustawiany dla wykrytego obiektu. Ma następujące właściwości:

Właściwość	Typ	Opis
discover	integer	Nadpisuje status discover dla wybranego obiektu. Możliwe wartości: 0 - Tak, kontynuuj wykrywanie obiektów; 1 - Nie, nowe obiekty nie będą wykrywane, a istniejące zostaną oznaczone jako utracone. Zachowanie właściwości: - *wymagane*

Okres operacji nadpisania reguły LLD

Okres operacji nadpisania reguły LLD to wartość interwału aktualizacji ustawiana dla wykrytej pozycji. Ma on następujące właściwości:

Property	Type	Description
delay	string	Nadpisuje interwał aktualizacji prototypu pozycji. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub, opcjonalnie, jeden lub więcej niestandardowych interwałów, rozdzielonych średnikami. Niestandardowe interwały mogą być kombinacją interwałów elastycznych i harmonogramowych. Akceptuje makra użytkownika lub makra LLD. Jeśli są używane, wartość musi być pojedynczym makrem. Wiele makr ani makra połączone z tekstem nie są obsługiwane. Interwały elastyczne można zapisać jako dwa makra rozdzielone ukośnikiem (np. `{$FLEX_INTERVAL}/{$FLEX_PERIOD}`). Przykład: `1h;wd1-5h9-18;{$Macro1}/1-7,00:00-24:00;0/6-7,12:00-24:00;{$Macr` Zachowanie właściwości: - *required*

Historia operacji nadpisania reguły LLD

Wartość historii operacji nadpisania reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
history	string	Nadpisuje historię prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane historyczne powinny być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *wymagane*

Trendy operacji nadpisania reguły LLD

Wartość trendów operacji nadpisania reguły LLD, która jest ustawiana dla wykrytej pozycji. Ma następujące właściwości:

Właściwość	Type	Opis
trends	string	Nadpisuje trendy prototypu pozycji, czyli jednostkę czasu określającą, jak długo dane trendów mają być przechowywane. Akceptuje również makro użytkownika i makro LLD. Zachowanie właściwości: - *required*

Ważność operacji nadpisania reguły LLD

Wartość ważności operacji nadpisania reguły LLD, która jest ustawiana dla odkrytego wyzwalacza. Ma następujące właściwości:

Właściwość	Typ	Opis
severity	integer	Nadpisuje ważność prototypu wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa. Zachowanie właściwości: - *wymagane*

Znacznik operacji nadpisania reguły LLD

Obiekt znacznika operacji nadpisania reguły LLD zawiera nazwę znacznika i wartość ustawiane dla wykrytego obiektu. Ma następujące właściwości:

Property	Type	Description
tag	string	Nowa nazwa znacznika. Akceptuje również makro użytkownika i makro LLD. Property behavior: - *required*
value	string	Nowa wartość znacznika. Akceptuje również makro użytkownika i makro LLD.

Szablon operacji przesłaniania reguły LLD

Obiekt szablonu operacji przesłaniania reguły LLD, który jest powiązany z wykrytym hostem. Ma następujące właściwości:

Właściwość	Type	Opis
templateid	ID	Przesłania szablon powiązanych szablonów prototypu hosta. Zachowanie właściwości: - *wymagane*

Wartość trybu inwentaryzacji operacji nadpisania reguły LLD

Wartość trybu inwentaryzacji operacji nadpisania reguły LLD, która jest ustawiana dla wykrytego hosta. Ma następujące właściwości:

Właściwość	Typ	Opis
inventory_mode	integer	Nadpisuje tryb inwentaryzacji prototypu hosta. Możliwe wartości: -1 - wyłączony; 0 - *(domyślnie)* ręczny; 1 - automatyczny. Zachowanie właściwości: - *wymagane*

discoveryrule.create

Opis

`object discoveryrule.create(object/array lldRules)`

Ta metoda umożliwia tworzenie nowych reguł LLD.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Reguły LLD do utworzenia.

Oprócz standardowych właściwości reguły LLD, metoda akceptuje następujące parametry.

Parametr	Type	Opis
filter	object	Filtr reguły LLD dla reguły LLD.
preprocessing	object/array	Opcje przetwarzania wstępnego reguły LLD.
lld_macro_paths	object/array	Opcje lld_macro_path reguły LLD.
overrides	object/array	Opcje nadpisań reguły LLD.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych reguł LLD we właściwości `itemids`. Kolejność zwróconych identyfikatorów ID jest zgodna z kolejnością przekazanych reguł LLD.

Przykłady

Tworzenie reguły LLD

Utwórz regułę LLD agenta Zabbix, aby wykrywać zamontowane systemy plików. Wykryte pozycje będą aktualizowane co 30 sekund.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Używanie filtra

Utwórz regułę LLD z zestawem warunków, według których będą filtrowane wyniki. Warunki zostaną zgrupowane przy użyciu operatora logicznego „and”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Filtered LLD rule",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
```

```
}
```

Tworzenie reguły LLD ze ścieżkami makr

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Reguła LLD ze ścieżkami makr 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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Użycie niestandardowego filtra wyrażeń

Utwórz regułę LLD z filtrem, który będzie używać niestandardowego wyrażenia do oceny warunków. Reguła LLD ma wykrywać tylko obiekty, których wartość makra ”{#MACRO1}” pasuje jednocześnie do obu wyrażeń regularnych ”regex1” i ”regex2”, a wartość ”{#MACRO2}” pasuje do ”regex3” lub ”regex4”. Identyfikatory formuł ”A”, ”B”, ”C” i ”D” zostały wybrane arbitralnie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Filtered LLD rule",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}
```

Używanie niestandardowych pól zapytania i nagłówków

Utwórz regułę LLD z niestandardowymi polami zapytania i nagłówkami.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
    "id": 1
}
```

Tworzenie reguły LLD z preprocessingiem

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Reguła wykrywania z preprocessingiem",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Tworzenie reguły LLD z nadpisaniami

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Discover database host",
        "key_": "lld.with.overrides",
        "hostid": "10001",
        "type": 0,
        "delay": "60s",
        "interfaceid": "1155",
        "overrides": [
            {
                "name": "Discover MySQL host",
                "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": "Discover PostgreSQL host",
                "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "30980"
        ]
    },
    "id": 1
}
```

Utwórz regułę LLD skryptu

Utwórz proste zbieranie danych za pomocą reguły LLD skryptu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Script example",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Utwórz regułę LLD z określonym okresem czasu wyłączania i bez usuwania

Utwórz regułę LLD z niestandardowym okresem czasu wyłączania encji po tym, jak nie jest już wykrywana, z ustawieniem, że nigdy nie zostanie usunięta.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

Zobacz także

- Filtr reguły LLD
- Ścieżki makr LLD
- Wstępne przetwarzanie reguły LLD

Źródło

CDiscoveryRule::create() w *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.delete

Opis

`object discoveryrule.delete(array lldRuleIds)`

Ta metoda umożliwia usuwanie reguł LLD.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.
 Użytkownik nie będzie mógł usunąć reguły LLD, która wykrywa encje za pomocą niedostępnego proxy. Więcej informacji można znaleźć w sekcji Konfiguracja grup użytkowników (*Lista dostępu do proxy*).

Parametry

`(array)` Identyfikatory IDs reguł LLD do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych reguł LLD we właściwości `ruleids`.

Przykłady

Usuwanie wielu reguł LLD

Usuń dwie reguły LLD.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "discoveryrule.delete",
"params": [
"27665",
"27668"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"ruleids": [
"27665",
"27668"
]
},
"id": 1
}
```

Źródło

CDiscoveryRule::delete() w *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.get

Opis

`integer/array discoveryrule.get(object parameters)`

Metoda umożliwia pobranie reguł LLD zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
itemids	ID/array	Zwraca tylko reguły LLD o podanych identyfikatorach.
groupids	ID/array	Zwraca tylko reguły LLD należące do hostów z podanych grup.
hostids	ID/array	Zwraca tylko reguły LLD należące do podanych hostów.
inherited	boolean	Jeśli ustawiono `true`, zwraca tylko reguły LLD odziedziczone z szablonu.
interfaceids	ID/array	Zwraca tylko reguły LLD używające podanych interfejsów hosta.
monitored	boolean	Jeśli ustawiono `true`, zwraca tylko włączone reguły LLD należące do monitorowanych hostów.
templated	boolean	Jeśli ustawiono `true`, zwraca tylko reguły LLD należące do szablonów.
templateids	ID/array	Zwraca tylko reguły LLD należące do podanych szablonów.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` z danymi obiektu wykrywania reguły LLD. Obiekt wykrywania reguły LLD łączy wykrytą regułę LLD z prototypem reguły LLD, z którego została wykryta.
selectDiscoveryRulePrototypes	query	Zwraca właściwość `discoveryRulePrototypes` z prototypami reguł LLD należącymi do reguły LLD. Obsługuje `count`.
selectFilter	query	Zwraca właściwość `filter` z danymi filtra używanego przez regułę LLD.
selectGraphs	query	Zwraca właściwość `graphs` z prototypami wykresów należącymi do reguły LLD. Obsługuje `count`.
selectHostPrototypes	query	Zwraca właściwość `hostPrototypes` z prototypami hostów należącymi do reguły LLD. Obsługuje `count`.
selectHosts	query	Zwraca właściwość `hosts` z tablicą hostów, do których należy reguła LLD.
selectItems	query	Zwraca właściwość `items` z prototypami pozycji należącymi do reguły LLD. Obsługuje `count`.
selectTriggers	query	Zwraca właściwość `triggers` z prototypami wyzwalaczy należącymi do reguły LLD. Obsługuje `count`.
selectLLDMacroPaths	query	Zwraca właściwość `lld_macro_paths` z listą makr LLD i ścieżek do wartości przypisanych do każdego odpowiadającego makra.
selectPreprocessing	query	Zwraca właściwość `preprocessing` z opcjami przetwarzania wstępnego reguły LLD.
selectOverrides	query	Zwraca właściwość `lld_rule_overrides` z listą filtrów nadpisywania, warunków i operacji wykonywanych na obiektach prototypów.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - nazwa techniczna hosta, do którego należy reguła LLD.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectItems`, `selectGraphs`, `selectTriggers`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `itemid`, `name`, `key_`, `delay`, `type`, `status`.

Parametr	Typ	Opis
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie reguł wykrywania z hosta

Pobierz wszystkie reguły wykrywania dla określonego ID hosta.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": "extend",
        "hostids": "10202"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "27425",
            "type": "0",
            "snmp_oid": "",
            "hostid": "10202",
            "name": "Wykrywanie interfejsów sieciowych",
            "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": "Wykrywanie interfejsów sieciowych zgodnie z definicją w globalnym wyrażeniu re
            "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": "Wykrywanie zamontowanych systemów plików",
            "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": "Wykrywanie systemów plików różnych typów zgodnie z definicją w globalnym wyraż
            "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
}
```

Pobieranie warunków filtra

Pobierz nazwę reguły LLD „24681” oraz jej warunki filtra. Filtr używa typu obliczania „and”, więc właściwość `formula` jest pusta, a `eval_formula` jest generowana automatycznie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "selectFilter": "extend",
        "itemids": ["24681"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobieranie reguły LLD według URL

Pobierz regułę LLD dla hosta na podstawie wartości pola URL reguły. Obsługiwane jest tylko dokładne dopasowanie ciągu URL zdefiniowanego dla reguły LLD.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "hostids": "10257",
        "filter": {
            "type": 19,
            "url": "http://127.0.0.1/discoverer.php"
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Pobieranie reguły LLD z nadpisaniami

Pobierz jedną regułę LLD, która ma różne ustawienia nadpisywania.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "itemids": "30980",
        "selectOverrides": ["name", "step", "stop", "filter", "operations"]
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "name": "Discover database host",
            "overrides": [
                {
                    "name": "Discover MySQL host",
                    "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": "Database host",
                            "opstatus": {
                                "status": "0"
                            },
                            "optag": [
                                {
                                    "tag": "database",
                                    "value": "mysql"
                                }
                            ],
                            "optemplate": [
                                {
                                    "templateid": "10170"
                                }
                            ]
                        }
                    ]
                },
                {
                    "name": "Discover PostgreSQL host",
                    "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": "Database host",
                            "opstatus": {
                                "status": "0"
                            },
                            "optag": [
                                {
                                    "tag": "database",
                                    "value": "postgresql"
                                }
                            ],
                            "optemplate": [
                                {
                                    "templateid": "10263"
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ],
    "id": 1
}
```


Zobacz także

- Prototyp wykresu
- Host
- Prototyp pozycji
- Filtr reguły LLD
- Prototyp wyzwalacza

Źródło

CDiscoveryRule::get() w *ui/include/classes/api/services/CDiscoveryRule.php*.

discoveryrule.update

Opis

`object discoveryrule.update(object/array lldRules)`

Ta metoda umożliwia aktualizację istniejących reguł LLD.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(object/array) Właściwości reguły LLD do zaktualizowania.

Właściwość itemid musi być zdefiniowana dla każdej reguły LLD, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości reguły LLD metoda akceptuje następujące parametry.

Parametr	Typ	Opis
filter	object	Filtr reguły LLD zastępujący istniejący filtr.
preprocessing	object/array	Opcje preprocessingu reguły LLD zastępujące istniejące opcje preprocessingu. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
lld_macro_paths	object/array	Opcje lld_macro_path reguły LLD zastępujące istniejące opcje lld_macro_path. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych
overrides	object/array	Opcje nadpisań reguły LLD zastępujące istniejące opcje nadpisań. Zachowanie parametru: - *tylko do odczytu* dla obiektów dziedziczonych

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych reguł LLD we właściwości itemids.

Przykłady

Dodawanie filtra do reguły LLD

Dodaj filtr, aby zawartość makra *{#FSTYPE}* odpowiadała wyrażeniu regularnemu *@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
}
```

Dodawanie ścieżek makr LLD

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "22450",
        "lld_macro_paths": [
            {
                "lld_macro": "{#MACRO1}",
                "path": "$.json.path"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "22450"
        ]
    },
    "id": 1
}
```

Wyłącz przechwytywanie

Wyłącz przechwytywanie LLD dla reguły wykrywania.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "28336",
        "allow_traps": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "28336"
        ]
    },
    "id": 1
}
```

Aktualizowanie opcji przetwarzania wstępnego reguły LLD

Zaktualizuj regułę LLD za pomocą reguły przetwarzania wstępnego „JSONPath”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}
```

Aktualizacja skryptu reguły LLD

Zaktualizuj skrypt reguły LLD innym skryptem i usuń zbędne parametry, które były używane przez poprzedni skrypt.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "23865",
        "parameters": [],
        "script": "Zabbix.log(3, 'Log test');\nreturn 1;"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "23865"
        ]
    },
    "id": 1
}
```

Aktualizacja czasu życia reguły LLD

Zaktualizuj regułę LLD, aby wyłączyć encję, która nie jest już wykrywana, po 12 godzinach i usunąć ją po 7 dniach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "discoveryrule.update",
    "params": {
        "itemid": "46864",
        "lifetime_type": 0,
        "lifetime": "7d",
        "enabled_lifetime_type": 0,
        "enabled_lifetime": "12h"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "46864"
        ]
    },
    "id": 1
}
```

Źródło

CDiscoveryRule::update() w *ui/include/classes/api/services/CDiscoveryRule.php*.

Reguła wykrywania

Ta klasa została zaprojektowana do pracy z regułami wykrywania sieci.

> **Note:**
> To API jest przeznaczone do pracy z regułami wykrywania sieci. W przypadku reguł wykrywania niskiego poziomu zobacz API reguł LLD.

Odwołania do obiektów:

- Reguła wykrywania

Dostępne metody:

- drule.create - utwórz nowe reguły wykrywania sieci
- drule.delete - usuń reguły wykrywania sieci
- drule.get - pobierz reguły wykrywania sieci
- drule.update - zaktualizuj reguły wykrywania sieci

Obiekt reguły odkrywania

Następujące obiekty są bezpośrednio powiązane z API `drule`.

Reguła discovery

Obiekt reguły discovery definiuje regułę wykrywania sieci. Ma on następujące właściwości.

Property	Type	Description
druleid	ID	Identyfikator reguły discovery. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* w operacjach aktualizacji
iprange	string	Jeden lub kilka zakresów adresów IP do sprawdzenia, rozdzielonych przecinkami. Więcej informacji o obsługiwanych formatach zakresów adresów IP można znaleźć w sekcji konfiguracja wykrywania sieci. Zachowanie właściwości: - *wymagana* w operacjach tworzenia
name	string	Nazwa reguły discovery. Zachowanie właściwości: - *wymagana* w operacjach tworzenia

Property	Type	Description
delay	string	Interwał wykonywania reguły discovery. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 2h, 1d) lub makro użytkownika. Wartość domyślna: 1h.
proxyid	ID	Identyfikator proxy używanego do wykrywania.
status	integer	Określa, czy reguła discovery jest włączona. Możliwe wartości: 0 - *(domyślnie)* włączona; 1 - wyłączona.
concurrency_max	integer	Maksymalna liczba równoczesnych sprawdzeń dla jednej reguły discovery. Możliwe wartości: 0 - *(domyślnie)* nieograniczona liczba sprawdzeń; 1 - jedno sprawdzenie; 2-999 - niestandardowa liczba sprawdzeń.
error	string	Tekst błędu, jeśli podczas wykonywania reguły discovery wystąpiły problemy. Zachowanie właściwości: - *tylko do odczytu*

drule.create

Opis

`object drule.create(object/array discoveryRules)`

Ta metoda umożliwia tworzenie nowych reguł wykrywania.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Reguły wykrywania do utworzenia.

Oprócz standardowych właściwości reguły wykrywania, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dchecks	array	Kontrole wykrywania do utworzenia dla reguły wykrywania. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych reguł wykrywania we właściwości `druleids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych reguł wykrywania.

Przykłady

Utwórz regułę wykrywania

Utwórz regułę wykrywania, aby znaleźć maszyny z uruchomionym agent Zabbix w sieci lokalnej. Reguła musi używać pojedynczego sprawdzenia agent Zabbix na porcie 10050.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "drule.create",
    "params": {
        "name": "Zabbix agent discovery",
        "iprange": "192.168.1.1-255",
        "concurrency_max": "10",
        "dchecks": [
            {
                "type": "9",
                "key_": "system.uname",
                "ports": "10050",
                "uniq": "0"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

Zobacz także

- Kontrola wykrywania

Źródło

CDRule::create() w *ui/include/classes/api/services/CDRule.php*.

drule.delete

Opis

`object drule.delete(array discoveryRuleIds)`

Ta metoda umożliwia usuwanie reguł wykrywania.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` Identyfikatory ID reguł wykrywania do usunięcia.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych reguł wykrywania w ramach właściwości `druleids`.

Przykłady

Usuń wiele reguł wykrywania

Usuń dwie reguły wykrywania.

Żądanie:

```
{
"jsonrpc": "2.0",
```

```
"method": "drule.delete",
"params": [
"4",
"6"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"druleids": [
"4",
"6"
]
},
"id": 1
}
```

Źródło

CDRule::delete() w *ui/include/classes/api/services/CDRule.php*.

drule.get

Opis

`integer/array drule.get(object parameters)`

Metoda umożliwia pobranie reguł wykrywania zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dhostids	ID/array	Zwraca tylko reguły wykrywania, które utworzyły podane wykryte hosty.
druleids	ID/array	Zwraca tylko reguły wykrywania o podanych ID.
dserviceids	ID/array	Zwraca tylko reguły wykrywania, które utworzyły podane wykryte usługi.
selectDChecks	query	Zwraca właściwość `dchecks` z kontrolami wykrywania używanymi przez regułę wykrywania. Obsługuje `count`.
selectDHosts	query	Zwraca właściwość `dhosts` z wykrytymi hostami utworzonymi przez regułę wykrywania. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Ma zastosowanie do następujących podzapytań: `selectDChecks` - wyniki będą sortowane według `dcheckid`; `selectDHosts` - wyniki będą sortowane według `dhostsid`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `druleid`, `name`.

Parametr	Typ	Opis
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobierz wszystkie reguły wykrywania

Pobierz wszystkie skonfigurowane reguły wykrywania oraz kontrole wykrywania, których używają.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "drule.get",
    "params": {
        "output": "extend",
        "selectDChecks": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Wykryty host
- Sprawdzenie wykrywania

Źródło

CDRule::get() w *ui/include/classes/api/services/CDRule.php*.

drule.update

Opis

`object drule.update(object/array discoveryRules)`

Ta metoda umożliwia aktualizację istniejących reguł wykrywania.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości reguły wykrywania do zaktualizowania.

Właściwość `druleid` musi być zdefiniowana dla każdej reguły wykrywania, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości reguły wykrywania, metoda akceptuje następujące parametry.

Parametr	Type	Opis
dchecks	array	Kontrole wykrywania zastępujące istniejące kontrole.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych reguł wykrywania właściwości `druleids`.

Przykłady

Zmień zakres IP reguły wykrywania

Zmień zakres IP reguły wykrywania na „192.168.2.1-255”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "drule.update",
    "params": {
        "druleid": "6",
        "iprange": "192.168.2.1-255"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "druleids": [
            "6"
        ]
    },
    "id": 1
}
```

Zobacz także

- Kontrola wykrywania

Źródło

CDRule::update() w *ui/include/classes/api/services/CDRule.php*.

Rola

Ta klasa jest przeznaczona do pracy z rolami użytkowników.

Odwołania do obiektów:

- Role
- Role rules
  - UI element
  - Service
  - Service tag
  - Module
  - Action

Dostępne metody:

- role.create - utwórz nowe role użytkowników
- role.delete - usuń role użytkowników
- role.get - pobierz role użytkowników
- role.update - zaktualizuj role użytkowników

Obiekt roli

Następujące obiekty są bezpośrednio powiązane z API `role`.

Rola

Obiekt roli ma następujące właściwości:

Właściwość	Typ	Opis
roleid	ID	ID roli. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa roli. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Typ użytkownika. Możliwe wartości: 1 - *(domyślnie)* Użytkownik; 2 - Admin; 3 - Super admin. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
readonly	integer	Czy rola jest tylko do odczytu. Możliwe wartości: 0 - *(domyślnie)* Nie; 1 - Tak. Zachowanie właściwości: - *tylko do odczytu*

Reguły roli

Obiekt reguł roli ma następujące właściwości:

Property	Type	Description
ui	array	Tablica obiektów elementów interfejsu użytkownika.
ui.default_access	integer	Określa, czy dostęp do nowych elementów interfejsu użytkownika jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
profile.redirect.enforce	integer	Określa, czy adres URL przekierowania na poziomie roli po uwierzytelnieniu jest wymuszany i nie może zostać zastąpiony przez użytkownika. Możliwe wartości: 0 - *(domyślnie)* Wyłączony; 1 - Włączony.
profile.redirect.url	string	Adres URL używany do przekierowania użytkownika po uwierzytelnieniu.
services.read.mode	integer	Dostęp do usług tylko do odczytu. Możliwe wartości: 0 - Dostęp tylko do odczytu do usług określonych przez właściwość `services.read.list` lub pasujących do właściwości `services.read.tag`; 1 - *(domyślnie)* Dostęp tylko do odczytu do wszystkich usług.
services.read.list	array	Tablica obiektów Service. Określone usługi, w tym usługi podrzędne, otrzymają dostęp tylko do odczytu dla roli użytkownika. Dostęp tylko do odczytu nie zastąpi dostępu do usług w trybie odczytu i zapisu. Zachowanie właściwości: - *obsługiwana*, jeśli `services.read.mode` ma wartość „0”.
services.read.tag	object	Tablica obiektów Service tag. Usługi pasujące do tagów, w tym usługi podrzędne, otrzymają dostęp tylko do odczytu dla roli użytkownika. Dostęp tylko do odczytu nie zastąpi dostępu do usług w trybie odczytu i zapisu. Zachowanie właściwości: - *obsługiwana*, jeśli `services.read.mode` ma wartość „0”.
services.write.mode	integer	Dostęp do usług w trybie odczytu i zapisu. Możliwe wartości: 0 - *(domyślnie)* Dostęp w trybie odczytu i zapisu do usług określonych przez właściwość `services.write.list` lub pasujących do właściwości `services.write.tag`; 1 - Dostęp w trybie odczytu i zapisu do wszystkich usług.
services.write.list	array	Tablica obiektów Service. Określone usługi, w tym usługi podrzędne, otrzymają dostęp w trybie odczytu i zapisu dla roli użytkownika. Dostęp w trybie odczytu i zapisu zastąpi dostęp tylko do odczytu do usług. Zachowanie właściwości: - *obsługiwana*, jeśli `services.write.mode` ma wartość „0”.

Property	Type	Description
services.write.tag	object	Tablica obiektów Service tag. Usługi pasujące do tagów, w tym usługi podrzędne, otrzymają dostęp w trybie odczytu i zapisu dla roli użytkownika. Dostęp w trybie odczytu i zapisu zastąpi dostęp tylko do odczytu do usług. Zachowanie właściwości: - *obsługiwana*, jeśli `services.write.mode` ma wartość „0”.
modules	array	Tablica obiektów modułów.
modules.default_access	integer	Określa, czy dostęp do nowych modułów jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
api.access	integer	Określa, czy dostęp do API jest włączony. Możliwe wartości: 0 - *(domyślnie)* Wyłączony; 1 - Włączony.
api.mode	integer	Tryb obsługi metod API wymienionych we właściwości `api`. Możliwe wartości: 0 - *(domyślnie)* Lista blokowanych; 1 - Lista dozwolonych. Zachowanie właściwości: - *obsługiwana*, jeśli `api.access` ma wartość „1”.
api	array	Tablica metod API. Obsługiwane są symbole wieloznaczne, na przykład: `*` (dowolna metoda), `dashboard.*`, `*.export`. Jeśli nie określono żadnych metod, dostęp do wszystkich metod API jest zabroniony, gdy `api.mode` ma wartość „1” (lista dozwolonych), i dozwolony, gdy `api.mode` ma wartość „0” (lista blokowanych). Zachowanie właściwości: - *obsługiwana*, jeśli `api.access` ma wartość „1”.
actions	array	Tablica obiektów akcji.
actions.default_access	integer	Określa, czy dostęp do nowych akcji jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.
devices.access	integer	Określa, czy dostęp do własnych urządzeń mobilnych użytkownika w celu korzystania z aplikacji mobilnej Zabbix jest włączony. Możliwe wartości: 0 - (*domyślnie dla użytkownika i administratora*) Wyłączony; 1 - (*domyślnie dla superadministratora*) Włączony. Zachowanie właściwości: - *obsługiwana*, jeśli `EnableMobileDevices` jest włączona
devices.actions	array	Tablica obiektów akcji urządzeń. Zachowanie właściwości: - *obsługiwana*, jeśli `devices.access` ma wartość „1”

Property	Type	Description
devices.actions.default_access	integer	Określa, czy dostęp do nowych akcji urządzeń jest włączony. Możliwe wartości: 0 - (*domyślnie dla użytkownika i administratora*) Wyłączony; 1 - (*domyślnie dla superadministratora*) Włączony. Zachowanie właściwości: - *obsługiwana*, jeśli `EnableMobileDevices` jest włączona

Element interfejsu użytkownika

Obiekt elementu interfejsu użytkownika ma następujące właściwości:

Property	Type	Description
name	string	Nazwa elementu interfejsu użytkownika. Możliwe wartości, jeśli właściwość type obiektu Role jest ustawiona na „User”, „Admin” lub „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*. Możliwe wartości, jeśli właściwość type obiektu Role jest ustawiona na „Admin” lub „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*. Możliwe wartości, jeśli właściwość type obiektu Role jest ustawiona na „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*. Zachowanie właściwości: *- required*

Property	Type	Description
status	integer	Określa, czy dostęp do elementu interfejsu użytkownika jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(default)* Włączony.

Usługa

Właściwość	Type	Opis
serviceid	ID	ID usługi. Property behavior: - *wymagane*

Znacznik usługi

Właściwość	Type	Opis
tag	string	Nazwa znacznika. Jeśli zostanie podany pusty ciąg znaków, znacznik usługi nie będzie używany do dopasowywania usług. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika. Jeśli nie zostanie podana żadna wartość lub zostanie podany pusty ciąg znaków, do dopasowywania usług będzie używana tylko nazwa znacznika.

Moduł

Obiekt modułu ma następujące właściwości:

Właściwość	Typ	Opis
moduleid	ID	ID modułu. Zachowanie właściwości: - *wymagane*
status	integer	Czy dostęp do modułu jest włączony. Możliwe wartości: 0 - Wyłączony; 1 - *(domyślnie)* Włączony.

Działanie

Obiekt działania ma następujące właściwości:

Property	Type	Description
name	string	Nazwa działania. Możliwe wartości, jeśli właściwość `type` obiektu Role jest ustawiona na „User”, „Admin” lub „Super admin”: `edit_dashboards` - Tworzenie i edytowanie pulpitów nawigacyjnych; `edit_maps` - Tworzenie i edytowanie map; `add_problem_comments` - Dodawanie komentarzy do problemów; `change_severity` - Zmienianie ważności problemu; `acknowledge_problems` - Potwierdzanie problemów; `suppress_problems` - Wyciszanie problemów; `close_problems` - Zamykanie problemów; `execute_scripts` - Wykonywanie skryptów; `manage_api_tokens` - Zarządzanie tokenami API; `change_problem_ranking` - Zmienianie klasyfikacji problemu z przyczyny na objaw i odwrotnie; `edit_own_media` - Zezwalanie użytkownikom na tworzenie i edytowanie własnych mediów. Możliwe wartości, jeśli właściwość `type` obiektu Role jest ustawiona na „Admin” lub „Super admin”: `edit_maintenance` - Tworzenie i edytowanie okresów konserwacji; `manage_scheduled_reports` - Zarządzanie zaplanowanymi raportami, `manage_sla` - Zarządzanie SLA, `select_server_for_monitoring` - Zezwalanie użytkownikom na tworzenie hostów monitorowanych przez serwer oraz przez reguły wykrywania sieci wykonywane przez serwer. Możliwe wartości, jeśli właściwość `type` obiektu Role jest ustawiona na „User” lub „Admin”: `invoke_execute_now` - Zezwalanie użytkownikom, którzy mają tylko uprawnienia do odczytu hosta, na wykonywanie kontroli pozycji. Możliwe wartości, jeśli właściwość `type` obiektu Role jest ustawiona na „Super admin”: `edit_user_media` - Zezwalanie użytkownikom na tworzenie i edytowanie mediów innych użytkowników. Zachowanie właściwości: *- required*
status	integer	Określa, czy dostęp do wykonywania działania jest włączony. Możliwe wartości: 0 - Wyłączone; 1 - *(default)* Włączone.

Działanie dotyczące urządzenia

Obiekt działania dotyczącego urządzenia ma następujące właściwości:

Property	Type	Description
name	string	Nazwa działania dotyczącego urządzenia. Możliwe wartości, jeśli właściwość `type` obiektu Role jest ustawiona na „User”, „Admin” lub „Super admin”: `manage_own` - Zezwalaj użytkownikom na dodawanie lub usuwanie własnych urządzeń mobilnych. Możliwe wartości, jeśli właściwość `type` obiektu Role jest ustawiona na „Super admin”: `manage_user` - Zezwalaj użytkownikom na dodawanie lub usuwanie urządzeń mobilnych wszystkich użytkowników. Zachowanie właściwości: - *required*
status	integer	Określa, czy dostęp do wykonywania działania jest włączony. Możliwe wartości: 0 - Wyłączone; 1 - (*default*) Włączone.

role.create

Opis

`object role.create(object/array roles)`

Ta metoda umożliwia tworzenie nowych ról.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników

Parametry

`(object/array)` Role do utworzenia.

Oprócz standardowych właściwości roli, metoda akceptuje następujące parametry.

Parametr	Type	Opis
rules	array	Reguły roli, które mają zostać utworzone dla roli.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych ról we właściwości `roleids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych ról.

Przykłady

Tworzenie roli

Utwórz rolę typu „Użytkownik” z zablokowanym dostępem do dwóch elementów interfejsu użytkownika.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Reguły roli
- Element interfejsu użytkownika
- Moduł
- Akcja

Źródło

CRole::create() w *ui/include/classes/api/services/CRole.php*.

role.delete

Opis

`object role.delete(array roleids)`

Ta metoda umożliwia usuwanie ról.

Note:
Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody można cofnąć w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(array)` ID ról do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych ról we właściwości `roleids`.

Przykłady

Usunięcie wielu ról użytkownika

Usunięcie dwóch ról użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.delete",
    "params": [
        "4",
```

```
        "5"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CRole::delete() w *ui/include/classes/api/services/CRole.php*.

role.get

Opis

`integer/array role.get(object parameters)`

Metoda umożliwia pobieranie ról zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
roleids	ID/array	Zwraca tylko role o podanych identyfikatorach.
current	boolean	Jeśli ustawiono wartość `true`, zwraca tylko rolę bieżącego użytkownika.
selectRules	query	Zwraca właściwość `rules` zawierającą reguły roli.
selectUsers	query	Zwraca właściwość `users` zawierającą użytkowników, którym przypisano tę rolę. Informacje o ograniczeniach zależnych od typu użytkownika znajdują się w opisie `user.get`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `roleid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	

Parametr	Typ	Opis
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych roli

Pobierz dane roli „Super admin role” oraz jej reguły dostępu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.get",
    "params": {
        "output": "extend",
        "selectRules": "extend",
        "roleids": "3"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Reguły roli
- Użytkownik

Źródło

CRole::get() w *ui/include/classes/api/services/CRole.php*.

role.update

Opis

`object role.update(object/array roles)`

Ta metoda umożliwia aktualizację istniejących ról.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości roli do zaktualizowania.

Właściwość `roleid` musi być zdefiniowana dla każdej roli, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości roli metoda akceptuje następujące parametry.

Parametr	Type	Opis
rules	array	Reguły dostępu rules do zaktualizowania dla roli.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych ról we właściwości `roleids`.

Przykłady

Wyłączanie możliwości wykonywania skryptów

Zaktualizuj rolę o ID „5”, wyłącz możliwość wykonywania skryptów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "actions": [
                    {
                        "name": "execute_scripts",
                        "status": "0"
                    }
                ]
            }
        }
    ],
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Ograniczanie dostępu do API

Zaktualizuj rolę o ID „5”, zabraniając wywoływania dowolnych metod „create”, „update” lub „delete”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "api.access": "1",
                "api.mode": "0",
                "api": ["*.create", "*.update", "*.delete"]
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CRole::update() w *ui/include/classes/api/services/CRole.php*.

Scenariusz WWW

Ta klasa jest przeznaczona do pracy ze scenariuszami WWW.

Odwołania do obiektów:

- Scenariusz WWW
- Krok scenariusza
  - Pole HTTP
- Tag scenariusza WWW

Dostępne metody:

- httptest.create - utwórz nowe scenariusze WWW
- httptest.delete - usuń scenariusze WWW

- httptest.get - pobierz scenariusze WWW
- httptest.update - zaktualizuj scenariusze WWW

Obiekt scenariusza internetowego

Następujące obiekty są bezpośrednio powiązane z API `webcheck`.

Scenariusz internetowy

Obiekt scenariusza internetowego ma następujące właściwości.

Property	Type	Description
httptestid	ID	Identyfikator scenariusza internetowego. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* w operacjach aktualizacji
hostid	ID	Identyfikator hosta, do którego należy scenariusz internetowy. Zachowanie właściwości: - *stałe* - *wymagane* w operacjach tworzenia
name	string	Nazwa scenariusza internetowego. Zachowanie właściwości: - *wymagane* w operacjach tworzenia
agent	string	Ciąg user agent, który będzie używany przez scenariusz internetowy. Wartość domyślna: Zabbix
authentication	integer	Metoda uwierzytelniania, która będzie używana przez scenariusz internetowy. Możliwe wartości: 0 - *(domyślnie)* brak; 1 - podstawowe uwierzytelnianie HTTP; 2 - uwierzytelnianie NTLM; 3 - uwierzytelnianie Kerberos; 4 - uwierzytelnianie Digest.
delay	string	Interwał wykonywania scenariusza internetowego. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`) lub makro użytkownika. Wartość domyślna: 1m.
headers	array	Nagłówki HTTP, które będą wysyłane podczas wykonywania żądania.
http_password	string	Hasło używane do podstawowego uwierzytelniania HTTP, uwierzytelniania NTLM, Kerberos lub Digest.
http_proxy	string	Proxy, który będzie używany przez scenariusz internetowy, w formacie *http://[username[:password]@]proxy.example.com[:port]*.
http_user	string	Nazwa użytkownika używana do podstawowego uwierzytelniania HTTP, uwierzytelniania NTLM, Kerberos lub Digest.
retries	integer	Liczba prób wykonania każdego kroku przez scenariusz internetowy przed zgłoszeniem błędu. Wartość domyślna: 1.
ssl_cert_file	string	Nazwa pliku certyfikatu SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_file	string	Nazwa pliku prywatnego klucza SSL używanego do uwierzytelniania klienta (musi być w formacie PEM).
ssl_key_password	string	Hasło prywatnego klucza SSL.

Property	Type	Description
status	integer	Określa, czy scenariusz internetowy jest włączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
templateid	ID	Identyfikator nadrzędnego scenariusza internetowego szablonu. Zachowanie właściwości: - *tylko do odczytu*
variables	array	Zmienne scenariusza internetowego.
verify_host	integer	Określa, czy należy sprawdzić, czy nazwa hosta używana w połączeniu jest zgodna z nazwą w certyfikacie hosta. Możliwe wartości: 0 - *(domyślnie)* pomiń weryfikację hosta; 1 - weryfikuj hosta.
verify_peer	integer	Określa, czy należy sprawdzić autentyczność certyfikatu hosta. Możliwe wartości: 0 - *(domyślnie)* pomiń weryfikację peer; 1 - weryfikuj peer.
uuid	string	Globalny unikatowy identyfikator używany do łączenia importowanych scenariuszy internetowych z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany. Zachowanie właściwości: - *obsługiwane*, jeśli scenariusz internetowy należy do szablonu

Krok scenariusza

Obiekt kroku scenariusza definiuje określone sprawdzenie scenariusza internetowego. Ma on następujące właściwości.

Property	Type	Description
name	string	Nazwa kroku scenariusza. Zachowanie właściwości: - *required*
no	integer	Numer sekwencyjny kroku w scenariuszu internetowym. Zachowanie właściwości: - *required*
url	string	Adres URL, który ma zostać sprawdzony. Zachowanie właściwości: - *required*
follow_redirects	integer	Określa, czy należy podążać za przekierowaniami HTTP. Możliwe wartości: 0 - nie podążać za przekierowaniami; 1 - *(default)* podążać za przekierowaniami.
headers	array	Nagłówki HTTP, które zostaną wysłane podczas wykonywania żądania. Nagłówki kroku scenariusza zastąpią nagłówki określone dla scenariusza internetowego.
posts	string/array	Zmienne HTTP POST jako ciąg znaków (surowe dane POST) lub tablica pól HTTP (dane pól formularza).
required	string	Tekst, który musi znajdować się w odpowiedzi.

Property	Type	Description
retrieve_mode	integer	Część odpowiedzi HTTP, którą krok scenariusza musi pobrać. Możliwe wartości: 0 - *(default)* tylko treść; 1 - tylko nagłówki; 2 - nagłówki i treść.
status_codes	string	Zakresy wymaganych kodów stanu HTTP, rozdzielone przecinkami.
timeout	string	Limit czasu żądania w sekundach. Akceptuje liczbę sekund (np. 1, 60, 300), wartość z przyrostkiem czasu (np. 30s, 1m, 1h) lub makro użytkownika. Wartość domyślna: 15s. Maksimum: 1h. Minimum: 1s.
variables	array	Zmienne kroku scenariusza.
query_fields	array	Pola zapytania - tablica pól HTTP, które zostaną dodane do adresu URL podczas wykonywania żądania.

Pole HTTP

Obiekt pola HTTP definiuje nazwę i wartość używane do określania zmiennych scenariusza WWW, nagłówków HTTP oraz pól POST lub pól zapytania. Ma następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa nagłówka/zmiennej/pola POST lub GET. Zachowanie właściwości: - *wymagane*
value	string	Wartość nagłówka/zmiennej/pola POST lub GET. Zachowanie właściwości: - *wymagane*

Tag scenariusza WWW

Obiekt tagu scenariusza WWW ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu scenariusza WWW. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu scenariusza WWW. Zachowanie właściwości: - *tylko do odczytu*.
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon; 1 - host. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

httptest.create

Opis

`object httptest.create(object/array webScenarios)`

Ta metoda umożliwia tworzenie nowych scenariuszy web.

Note:
Utworzenie scenariusza web spowoduje automatyczne utworzenie zestawu pozycji monitorowania web.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Scenariusze web do utworzenia.

Oprócz standardowych właściwości scenariusza web, metoda akceptuje następujące parametry.

Parametr	Type	Opis
steps	array	Kroki scenariusza. Zachowanie parametru: - *wymagany*
tags	array	Tagi scenariusza web.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych scenariuszy WWW we właściwości `httptestids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych scenariuszy WWW.

Przykłady

Tworzenie scenariusza WWW

Utwórz scenariusz WWW do monitorowania strony głównej firmy. Scenariusz będzie miał dwa kroki, aby sprawdzić stronę główną oraz stronę „O nas” i upewnić się, że zwracają kod statusu HTTP 200.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Krok scenariusza

Źródło

CHttpTest::create() w *ui/include/classes/api/services/CHttpTest.php*.

httptest.delete

Opis

`object httptest.delete(array webScenarioIds)`

Ta metoda umożliwia usuwanie scenariuszy web.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID scenariuszy sieci WWW do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych scenariuszy sieci WWW we właściwości `httptestids`.

Przykłady

Usuwanie wielu scenariuszy sieci WWW

Usuń dwa scenariusze sieci WWW.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "httptest.delete",
"params": [
"2",
"3"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"httptestids": [
"2",
"3"
]
},
"id": 1
}
```

Źródło

CHttpTest::delete() w *ui/include/classes/api/services/CHttpTest.php*.

httptest.get

Opis

`integer/array httptest.get(object parameters)`

Metoda umożliwia pobieranie scenariuszy webowych zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwraca tylko scenariusze WWW należące do podanych grup hostów.
hostids	ID/array	Zwraca tylko scenariusze WWW należące do podanych hostów.
httptestids	ID/array	Zwraca tylko scenariusze WWW o podanych identyfikatorach.
inherited	boolean	Jeśli ustawiono na `true`, zwraca tylko scenariusze WWW odziedziczone z szablonu.
inheritedTags	boolean	Zwraca scenariusze WWW, które mają podane `tags` również w szablonie/hoście/podłączonych szablonach. Możliwe wartości: `true` - szablon/host/podłączone szablony również muszą mieć podane tagi; `false` - *(domyślnie)* tagi z szablonu/hosta/podłączonych szablonów są ignorowane.
monitored	boolean	Jeśli ustawiono na `true`, zwraca tylko włączone scenariusze WWW należące do monitorowanych hostów.
selectInheritedTags	query	Zwraca właściwość `inheritedTags` z tagami znajdującymi się w szablonie/hoście/podłączonych szablonach.
templated	boolean	Jeśli ustawiono na `true`, zwraca tylko scenariusze WWW należące do szablonów.
templateids	ID/array	Zwraca tylko scenariusze WWW należące do podanych szablonów.
expandName	flag	Rozwija makra w nazwie scenariusza WWW.
expandStepName	flag	Rozwija makra w nazwach kroków scenariusza.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwraca tylko scenariusze WWW z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie scenariusze WWW. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równe; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
selectHosts	query	Zwraca hosty, do których należy scenariusz WWW, jako tablicę we właściwości `hosts`.

Parametr	Typ	Opis
selectSteps	query	Zwraca kroki scenariusza WWW we właściwości `steps`. Obsługuje `count`.
selectTags	query	Zwraca tagi scenariusza WWW we właściwości `tags`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `httptestid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie scenariusza WWW

Pobierz wszystkie dane o scenariuszu WWW „9”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.get",
    "params": {
        "output": "extend",
        "selectSteps": "extend",
        "httptestids": "9"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- Host
- Krok scenariusza

Źródło

CHttpTest::get() w *ui/include/classes/api/services/CHttpTest.php*.

httptest.update

Opis

`object httptest.update(object/array webScenarios)`

Ta metoda umożliwia aktualizację istniejących scenariuszy web.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości scenariusza WWW do zaktualizowania.

Właściwość `httptestid` musi być zdefiniowana dla każdego scenariusza WWW, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości scenariusza WWW, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
steps	array	Kroki scenariusza zastępujące istniejące kroki.
tags	array	Tagi scenariusza WWW.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych scenariuszy WWW we właściwości `httptestid`.

Przykłady

Włączanie scenariusza WWW

Włącz scenariusz WWW, to znaczy ustaw jego status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "httptest.update",
    "params": {
        "httptestid": "5",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "httptestids": [
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Krok scenariusza

Źródło

CHttpTest::update() w *ui/include/classes/api/services/CHttpTest.php*.

Skrypt

Ta klasa jest przeznaczona do pracy ze skryptami.

Odwołania do obiektów:

- Script
  - Parametry webhooka
- Debug
  - Wpis dziennika

Dostępne metody:

- script.create - utwórz nowe skrypty
- script.delete - usuń skrypty
- script.execute - uruchom skrypty
- script.get - pobierz skrypty
- script.getscriptsbyevents - pobierz skrypty dla zdarzeń
- script.getscriptsbyhosts - pobierz skrypty dla hostów
- script.update - zaktualizuj skrypty

Obiekt skryptu

Następujące obiekty są bezpośrednio powiązane z API `script`.

Skrypt

Obiekt skryptu ma następujące właściwości.

Property	Type	Description
scriptid	ID	Identyfikator skryptu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa skryptu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
type	integer	Typ skryptu. Możliwe wartości, jeśli `scope` jest ustawione na „action operation”: 0 - Skrypt; 1 - IPMI; 2 - SSH; 3 - TELNET; 5 - Webhook. Możliwe wartości, jeśli `scope` jest ustawione na „manual host action” lub „manual event action”: 6 - URL. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
command	string	Polecenie do uruchomienia. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Script”, „IPMI”, „SSH”, „TELNET” lub „Webhook”
scope	integer	Zakres skryptu. Możliwe wartości: 1 - action operation; 2 - manual host action; 4 - manual event action. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Property	Type	Description
execute_on	integer	Miejsce uruchomienia skryptu. Możliwe wartości: 0 - uruchomienie na agencie Zabbix; 1 - uruchomienie na serwerze Zabbix. Jest *obsługiwane* tylko wtedy, gdy na serwerze Zabbix włączono wykonywanie skryptów globalnych; 2 - *(domyślnie)* uruchomienie na serwerze Zabbix lub proxy. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Script”
menu_path	string	Foldery rozdzielone ukośnikami, które po kliknięciu hosta lub zdarzenia tworzą w frontend nawigację przypominającą menu. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na „manual host action” lub „manual event action”
authtype	integer	Metoda uwierzytelniania używana dla skryptu typu SSH. Możliwe wartości: 0 - hasło; 1 - klucz publiczny. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „SSH”
username	string	Nazwa użytkownika używana do uwierzytelniania. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „SSH” lub „TELNET”
password	string	Hasło używane w skryptach SSH z uwierzytelnianiem za pomocą hasła oraz w skryptach TELNET. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „SSH”, a `authtype` na „password”, lub jeśli `type` jest ustawione na „TELNET”
publickey	string	Nazwa pliku klucza publicznego używanego w skryptach SSH z uwierzytelnianiem za pomocą klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „SSH”, a `authtype` na „public key”
privatekey	string	Nazwa pliku klucza prywatnego używanego w skryptach SSH z uwierzytelnianiem za pomocą klucza publicznego. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „SSH”, a `authtype` na „public key”
port	string	Numer portu używany w skryptach SSH i TELNET. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „SSH” lub „TELNET”
groupid	ID	Identyfikator grupy hostów, na których można uruchomić skrypt. Jeśli ustawiono wartość „0”, skrypt będzie dostępny we wszystkich grupach hostów. Wartość domyślna: 0.

Property	Type	Description
usrgrpid	ID	Identyfikator grupy użytkowników, która będzie mogła uruchamiać skrypt. Jeśli ustawiono wartość „0”, skrypt będzie dostępny dla wszystkich grup użytkowników. Wartość domyślna: 0. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na „manual host action” lub „manual event action”
host_access	integer	Uprawnienia do hosta wymagane do uruchomienia skryptu. Możliwe wartości: 2 - *(domyślnie)* odczyt; 3 - zapis. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na „manual host action” lub „manual event action”
confirmation	string	Tekst okna potwierdzenia. Okno pojawi się podczas próby uruchomienia skryptu z frontend Zabbix. Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na „manual host action” lub „manual event action”
timeout	string	Limit czasu wykonywania skryptu webhook w sekundach. Obsługiwane są przyrostki czasu (np. `30s`, `1m`). Możliwe wartości: 1-60s. Wartość domyślna: 30s. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „Webhook”
parameters	array	Tablica parametrów wejściowych webhook. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Webhook”
description	string	Opis skryptu.
url	string	URL zdefiniowany przez użytkownika. Zachowanie właściwości: - *wymagane*, jeśli `type` jest ustawione na „URL”
new_window	integer	Otwieranie URL w nowym oknie. Możliwe wartości: 0 - Nie; 1 - *(domyślnie)* Tak. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „URL”

Property	Type	Description
manualinput	integer	Określa, czy skrypt akceptuje dane wejściowe podane przez użytkownika. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone; Zachowanie właściwości: - *obsługiwane*, jeśli `scope` jest ustawione na „manual host action” lub „manual event action”
manualinput_prompt	string	Tekst monitu o dane wejściowe. Zachowanie właściwości: - *wymagane*, jeśli `manualinput` jest ustawione na „Enabled”
manualinput_validator	string	Pole tekstowe używane do sprawdzania danych wejściowych podanych przez użytkownika. Ciąg składa się z wyrażenia regularnego lub zestawu wartości rozdzielonych przecinkami. Zachowanie właściwości: - *wymagane*, jeśli `manualinput` jest ustawione na „Enabled”
manualinput_validator_type	integer	Określa oczekiwany typ danych wejściowych użytkownika. Możliwe wartości: 0 - *(domyślnie)* Ciąg. Wskazuje, że manualinput_validator należy traktować jako wyrażenie regularne; 1 - Lista. Wskazuje, że manualinput_validator należy traktować jako rozdzielaną przecinkami listę możliwych wartości wejściowych. Zachowanie właściwości: - *obsługiwane*, jeśli `manualinput` jest ustawione na „Enabled”
manualinput_default_value	string	Wartość domyślna używana do automatycznego wypełniania danych wejściowych użytkownika. Zachowanie właściwości: - *obsługiwane*, jeśli `manualinput_validator_type` jest ustawione na „String”

Parametry webhooka

Parametry przekazywane do skryptu webhooka podczas jego wywołania mają następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru. Obsługuje makra.

Debug

Informacje debugowania wykonanego skryptu webhook. Obiekt debug ma następujące właściwości.

Właściwość	Type	Opis
logs	array	Tablica wpisów dziennika.
ms	string	Czas wykonywania skryptu w milisekundach.

Wpis dziennika

Obiekt wpisu dziennika ma następujące właściwości.

Właściwość	Typ	Opis
level	integer	Poziom dziennika.
ms	string	Czas, który upłynął w milisekundach od uruchomienia skryptu do momentu dodania wpisu dziennika.
message	string	Komunikat dziennika.

script.create

Opis

`object script.create(object/array scripts)`

Ta metoda umożliwia tworzenie nowych skryptów.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Skrypty do utworzenia.

Metoda akceptuje skrypty ze standardowymi właściwościami skryptu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych skryptów we właściwości `scriptids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych skryptów.

Przykłady

Utwórz skrypt webhook

Utwórz skrypt webhook, który wysyła żądanie HTTP do zewnętrznej usługi.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "3"
        ]
    },
    "id": 1
}
```

Utwórz skrypt SSH

Utwórz skrypt SSH z uwierzytelnianiem kluczem publicznym, który może być wykonywany na hoście i ma menu kontekstowe.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "5"
        ]
    },
    "id": 1
}
```

Utwórz niestandardowy skrypt

Utwórz niestandardowy skrypt, który zrestartuje serwer. Skrypt będzie wymagał uprawnień do zapisu do hosta i poprosi użytkownika o ręczne wprowadzenie danych. Po pomyślnym przesłaniu danych wejściowych skrypt wyświetli komunikat potwierdzający we frontendzie.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "4"
        ]
    },
    "id": 1
}
```

Utwórz skrypt typu URL

Utwórz skrypt typu URL dla zakresu host, który pozostaje w tym samym oknie i ma tekst potwierdzenia.

Żądanie:

```
{
    "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": "Edytować host {HOST.NAME}?",
        "new_window": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "56"
        ]
    },
    "id": 1
}
```

Utwórz skrypt typu URL z ręcznym wprowadzaniem

Utwórz skrypt typu URL dla zakresu zdarzenia, który otwiera się w nowym oknie i ma ręczne wprowadzanie.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Skrypt URL z ręcznym wprowadzaniem",
        "type": 6,
        "scope": 4,
        "url": "http://zabbix/ui/zabbix.php?action={MANUALINPUT}",
        "new_window": 1,
        "manualinput": 1,
```

```
        "manualinput_prompt": "Wybierz stronę do otwarcia:",
        "manualinput_validator": "dashboard.view,script.list,actionlog.list",
        "manualinput_validator_type": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "57"
        ]
    },
    "id": 1
}
```

Źródło

CScript::create() w *ui/include/classes/api/services/CScript.php*.

script.delete

Opis

`object script.delete(array scriptIds)`

Ta metoda umożliwia usuwanie skryptów.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID skryptów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych skryptów we właściwości `scriptids`.

Przykłady

Usuń wiele skryptów

Usuń dwa skrypty.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "script.delete",
"params": [
"3",
"4"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"scriptids": [
"3",
"4"
```

```
]
},
"id": 1
}
```

Źródło

CScript::delete() w *ui/include/classes/api/services/CScript.php*.

script.execute

Opis

`object script.execute(object parameters)`

Ta metoda umożliwia uruchomienie skryptu na host lub zdarzeniu. Wyjątek stanowią skrypty typu URL; nie można ich wykonywać.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające ID skryptu do uruchomienia oraz albo ID hosta, albo ID zdarzenia i wartość manualinput.

Parametr	Type	Opis
scriptid	ID	ID skryptu do uruchomienia. Zachowanie parametru: - *wymagany*
hostid	ID	ID hosta, na którym ma zostać uruchomiony skrypt. Zachowanie parametru: - *wymagany*, jeśli `eventid` nie jest ustawione
eventid	ID	ID zdarzenia, dla którego ma zostać uruchomiony skrypt. Zachowanie parametru: - *wymagany*, jeśli `hostid` nie jest ustawione
manualinput	string	Wartość podana przez użytkownika, z którą ma zostać uruchomiony skrypt, zastępująca makro {MANUALINPUT}.

Zwracane wartości

`(object)` Zwraca wynik wykonania skryptu.

Właściwość	Type	Opis
response	string	Czy skrypt został uruchomiony pomyślnie. Możliwa wartość - `success`.
value	string	Dane wyjściowe skryptu.
debug	object	Zawiera obiekt debug, jeśli wykonywany jest skrypt webhook. Dla innych typów skryptów zawiera pusty obiekt.

Przykłady

Uruchamianie skryptu webhook

Uruchom skrypt webhook, który wysyła żądanie HTTP do zewnętrznej usługi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "4",
        "hostid": "30079"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Uruchamianie niestandardowego skryptu

Uruchom skrypt „ping” na hoście.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
    "params": {
        "scriptid": "1",
        "hostid": "30079"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Uruchom własny skrypt z ręcznym wprowadzaniem danych

Uruchom skrypt „ping” z poleceniem „ping -c {MANUALINPUT} {HOST.CONN}; case $? in [01]) true;; *) false;; esac” na hoście.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.execute",
```

```
    "params": {
        "scriptid": "7",
        "hostid": "30079",
        "manualinput": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CScript::execute() w *ui/include/classes/api/services/CScript.php*.

script.get

Opis

`integer/array script.get(object parameters)`

Metoda umożliwia pobieranie skryptów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
groupids	ID/array	Zwraca tylko skrypty, które mogą być uruchamiane dla podanych grup hostów.
hostids	ID/array	Zwraca tylko skrypty, które mogą być uruchamiane dla podanych hostów.
scriptids	ID/array	Zwraca tylko skrypty o podanych ID.
usrgrpids	ID/array	Zwraca tylko skrypty, które mogą być uruchamiane przez użytkowników z podanych grup użytkowników.
selectHostGroups	query	Zwraca właściwość `hostgroups` z grupami hostów, dla których skrypt może być uruchamiany.
selectHosts	query	Zwraca właściwość `hosts` z hostami, dla których skrypt może być uruchamiany.
selectActions	query	Zwraca właściwość `actions` z akcjami, z którymi skrypt jest powiązany.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `scriptid`, `name`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	

Parametr	Typ	Opis
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobierz wszystkie skrypty

Pobierz wszystkie skonfigurowane skrypty.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "Select a page to open:",
            "manualinput_validator": "dashboard.view,script.list,actionlog.list",
            "manualinput_validator_type": "1",
            "parameters": []
        }
    ],
    "id": 1
}
```


Zobacz także

- Host
- Grupa hostów

Źródło

CScript::get() w *ui/include/classes/api/services/CScript.php*.

script.getscriptsbyevents

Opis

`object script.getscriptsbyevents(object parameters)`

Ta metoda umożliwia pobranie wszystkich dostępnych skryptów dla podanego zdarzenia lub konkretnego skryptu, jeśli podano ID skryptu. Gdy podano manualinput, metoda zastępuje makro `{MANUALINPUT}` określoną wartością.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Metoda akceptuje obiekt lub tablicę obiektów z następującymi parametrami.

Parametr	Type	Opis
eventid	ID	ID zdarzenia, dla którego mają zostać zwrócone skrypty. Musi być unikalne. Zachowanie parametrów: - *wymagany*
scriptid	ID	ID skryptu do zwrócenia.
manualinput	string	Wartość podanej przez użytkownika wartości makra {MANUALINPUT}.

Zwracane wartości

`(object)` Zwraca obiekt z identyfikatorami zdarzeń jako właściwościami oraz tablicami dostępnych skryptów jako wartościami. Jeśli podano identyfikator skryptu, skojarzona wartość jest tablicą zawierającą określony skrypt.

Note:
Metoda automatycznie rozwinie makra w tekście `confirmation`, tekście `manualinput prompt` oraz w `url`.
Jeśli parametr manualinput zostanie podany, makro {MANUALINPUT} zostanie rozwinięte do określonej wartości.

Przykłady

Pobieranie skryptów według identyfikatorów zdarzeń

Pobierz wszystkie skrypty dostępne dla zdarzeń „632” i „614”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
         "eventid":  "632"
      },
      {
         "eventid":  "614"
      }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "632": [
            {
                "scriptid": "3",
                "name": "Wykryj system operacyjny",
                "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": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *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": "Strona Zabbix do otwarcia:",
            "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": "Wykryj system operacyjny",
            "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": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *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": "Strona Zabbix do otwarcia:",
            "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
}
```

Pobieranie określonego skryptu z wartością manualinput.

Pobierz skrypt o ID „4” dla zdarzenia „632” z wartością manualinput „dashboard.view”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyevents",
    "params": [
      {
        "eventid":  "632",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
      "632": [
        {
          "scriptid": "4",
          "name": "Open Zabbix page",
          "command": "",
          "host_access": "2",
          "usrgrpid": "0",
```

```
            "groupid": "0",
            "description": "",
            "confirmation": "Are you sure you want to open page 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": "Zabbix page to open:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        }
      ]
    },
    "id": 1
}
```


Źródło

CScript::getScriptsByEvents() w *ui/include/classes/api/services/CScript.php*.

script.getscriptsbyhosts

Opis

`object script.getscriptsbyhosts(object parameters)`

Ta metoda umożliwia pobranie wszystkich dostępnych skryptów dla wskazanego hosta lub konkretnego skryptu, jeśli podano identyfikator skryptu. Gdy podano manualinput, zastępuje ona makro {MANUALINPUT} określoną wartością.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object/array)` Metoda akceptuje obiekt lub tablicę obiektów z następującymi parametrami.

Parametr	Typ	Opis
hostid	ID	ID hosta, dla którego mają zostać zwrócone skrypty. Musi być unikalne. Zachowanie parametrów: - *wymagane*
scriptid	ID	ID skryptu do zwrócenia.
manualinput	string	Wartość makra {MANUALINPUT} podana przez użytkownika.

Zwracane wartości

`(object)` Zwraca obiekt z identyfikatorami hostów jako właściwościami oraz tablicami dostępnych skryptów jako wartościami. Jeśli podano identyfikator skryptu, skojarzona wartość jest tablicą zawierającą określony skrypt.

Note:
Metoda automatycznie rozwinie makra w tekście `confirmation`, tekście `manualinput prompt` oraz w `url`.
Jeśli parametr manualinput zostanie podany, makro {MANUALINPUT} zostanie rozwinięte do określonej wartości.

Przykłady

Pobieranie skryptów według identyfikatorów hostów

Pobierz wszystkie skrypty dostępne na hostach „30079” i „30073”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
       {
          "hostid":  "30079"
       },
       {
          "hostid":  "30073"
       }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "30079": [
            {
                "scriptid": "3",
                "name": "Wykryj system operacyjny",
                "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": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *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": "Strona Zabbix do otwarcia:",
            "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": "Wykryj system operacyjny",
            "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": "Otwórz stronę Zabbix",
            "command": "",
            "host_access": "2",
            "usrgrpid": "0",
            "groupid": "0",
            "description": "",
            "confirmation": "Czy na pewno chcesz otworzyć stronę *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": "Strona Zabbix do otwarcia:",
            "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
}
```

Pobierz określony skrypt z wartością manualinput.

Pobierz skrypt o ID „4” na hoście „30079” z wartością manualinput „dashboard.view”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.getscriptsbyhosts",
    "params": [
      {
        "hostid":  "30079",
        "scriptid": "4",
        "manualinput": "dashboard.view"
      }
  ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
      "30079": [
        {
          "scriptid": "4",
          "name": "Open Zabbix page",
          "command": "",
          "host_access": "2",
          "usrgrpid": "0",
          "groupid": "0",
          "description": "",
          "confirmation": "Are you sure you want to open page 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": "Zabbix page to open:",
            "manualinput_validator_type": "1",
            "manualinput_validator": "dashboard.view,discovery.view",
            "manualinput_default_value": "",
            "parameters": []
        }
      ]
    },
    "id": 1
}
```

Źródło

CScript::getScriptsByHosts() w *ui/include/classes/api/services/CScript.php*.

script.update

Opis

`object script.update(object/array scripts)`

Ta metoda umożliwia aktualizację istniejących skryptów.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości skryptu, które mają zostać zaktualizowane.

Właściwość `scriptid` musi być zdefiniowana dla każdego skryptu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian. Wyjątkiem jest zmiana właściwości `type` z 5 (Webhook) na inną: właściwość `parameters` zostanie wyczyszczona.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych skryptów we właściwości `scriptids`.

Przykłady

Zmień polecenie skryptu

Zmień polecenie skryptu na "/bin/ping -c 10 {HOST.CONN} 2>&1".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
        "scriptid": "1",
        "command": "/bin/ping -c 10 {HOST.CONN} 2>&1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
```

```
    "id": 1
}
```

Zmień polecenie skryptu i dodaj ręczne dane wejściowe

Zmień polecenie skryptu na "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1".

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "script.update",
    "params": {
      "scriptid": "1",
      "command": "/bin/ping -c {MANUALINPUT} {HOST.CONN} 2>&1",
      "manualinput": "1",
      "manualinput_prompt": "Określ liczbę pakietów ICMP do wysłania za pomocą polecenia ping",
      "manualinput_validator": "^(?:[1-9]|10)$",
      "manualinput_validator_type": "0",
      "manualinput_default_value": "10"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CScript::update() w *ui/include/classes/api/services/CScript.php*.

SLA

Ta klasa została zaprojektowana do pracy z obiektami SLA (Service Level Agreement) używanymi do oceny wydajności infrastruktury IT i usług biznesowych.

Odwołania do obiektów:

- SLA
- Harmonogram SLA
- Wykluczony przestój SLA
- Tag usługi SLA

Dostępne metody:

- sla.create - tworzenie nowych SLA
- sla.delete - usuwanie SLA
- sla.get - pobieranie SLA
- sla.getsli - pobieranie danych Service Level Indicator (SLI) dla SLA
- sla.update - aktualizowanie SLA

Obiekt SLA (Service Level Agreement)

Następujące obiekty są bezpośrednio powiązane z API `sla` (Service Level Agreement).

SLA

Obiekt SLA ma następujące właściwości.

Właściwość	Typ	Opis
slaid	ID	ID SLA. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa SLA. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
period	integer	Okres raportowania SLA. Możliwe wartości: 0 - dziennie; 1 - tygodniowo; 2 - miesięcznie; 3 - kwartalnie; 4 - rocznie. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
slo	float	Minimalny akceptowalny cel poziomu usług (Service Level Objective, SLO) wyrażony w procentach. Jeśli wskaźnik poziomu usług (Service Level Indicator, SLI) spadnie poniżej tej wartości, SLA jest uznawane za będące w stanie problemu/niespełnienia. Możliwe wartości: 0-100 (do 4 cyfr po przecinku). Zachowanie właściwości: - *wymagane* dla operacji tworzenia
effective_date	integer	Data wejścia SLA w życie. Możliwe wartości: znacznik czasu daty w UTC.
timezone	string	Strefa czasowa raportowania, na przykład: `Europe/London`, `UTC`. Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
status	integer	Status SLA. Możliwe wartości: 0 - *(domyślnie)* SLA wyłączone; 1 - SLA włączone.
description	string	Opis SLA.

Harmonogram SLA

Obiekt harmonogramu SLA definiuje okresy, w których połączone usługi mają działać prawidłowo. Ma następujące właściwości.

Właściwość	Type	Opis
period_from	integer	Czas rozpoczęcia cyklicznego tygodniowego okresu czasu (włącznie). Możliwe wartości: liczba sekund (licząc od niedzieli). Zachowanie właściwości: - *wymagane*

Właściwość	Type	Opis
period_to	integer	Czas zakończenia cyklicznego tygodniowego okresu czasu (wyłącznie). Możliwe wartości: liczba sekund (licząc od niedzieli). Zachowanie właściwości: - *wymagane*

Wyłączony czas niedostępności SLA

Obiekt wyłączonego czasu niedostępności definiuje okresy, w których powiązana usługa lub usługi mają być planowo wyłączone z działania, bez wpływu na SLI, np. podczas planowanej konserwacji. Ma następujące właściwości.

Właściwość	Type	Opis
name	string	Nazwa wykluczonego okresu niedostępności. Zachowanie właściwości: - *wymagane*
period_from	integer	Czas rozpoczęcia wykluczonego okresu niedostępności (włącznie). Możliwe wartości: znacznik czasu. Zachowanie właściwości: - *wymagane*
period_to	integer	Czas zakończenia wykluczonego okresu niedostępności (wyłącznie). Możliwe wartości: znacznik czasu. Zachowanie właściwości: - *wymagane*

Znacznik usługi SLA

Obiekt znacznika usługi SLA łączy usługi, które mają zostać uwzględnione w obliczeniach SLA. Ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa tagu usługi SLA. Zachowanie właściwości: - *wymagane*
operator	integer	Operator tagu usługi SLA. Możliwe wartości: 0 - *(domyślnie)* Równa się; 2 - Zawiera.
value	string	Wartość tagu usługi SLA.

sla.create

Opis

`object sla.create(object/array SLAs)`

Ta metoda umożliwia tworzenie nowych obiektów SLA.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(object/array) Obiekty SLA do utworzenia.

Oprócz standardowych właściwości SLA, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
service_tags	array	Tagi usług SLA, które mają zostać utworzone dla SLA. Zachowanie parametrów: - *wymagane*
schedule	array	Harmonogram SLA, który ma zostać utworzony dla SLA. Podanie pustego parametru będzie interpretowane jako harmonogram 24x7. Domyślnie: harmonogram 24x7.
excluded_downtimes	array	Wykluczone przestoje SLA, które mają zostać utworzone dla SLA.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory utworzonych SLA we właściwości slaids. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych SLA.

Przykłady

Tworzenie SLA

Poleć utworzenie wpisu SLA dla: * śledzenia dostępności usług związanych z silnikiem SQL; * niestandardowego harmonogramu obejmującego wszystkie dni robocze, z wyłączeniem ostatniej godziny w sobotę; * daty wejścia w życie przypadającej na ostatni dzień roku 2022; * planowanego przestoju trwającego 1 godzinę i 15 minut, rozpoczynającego się o północy 4 lipca; * włączonego obliczania tygodniowego raportu SLA; * minimalnego akceptowalnego SLO na poziomie 99,9995%.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.create",
    "params": [
        {
            "name": "Czas działania bazy danych",
            "slo": "99.9995",
            "period": "1",
            "timezone": "America/Toronto",
            "description": "Zapewnienie doskonałej dostępności głównych silników baz danych.",
            "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": "Wdrożenie aktualizacji wersji oprogramowania",
                    "period_from": "1648760400",
```

```
                    "period_to": "1648764900"
                }
            ]
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CSla::create() w *ui/include/classes/api/services/CSla.php*.

sla.delete

Opis

`object sla.delete(array slaids)`

Ta metoda umożliwia usuwanie wpisów SLA.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID SLA do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych SLA we właściwości `slaids`.

Przykłady

Usuwanie wielu SLA

Usuń dwa wpisy SLA.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "4",
```

```
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CSla::delete() w *ui/include/classes/api/services/CSla.php*.

sla.get

Opis

`integer/array sla.get(object parameters)`

Metoda umożliwia pobieranie obiektów SLA zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
slaids	ID/array	Zwraca tylko SLA o podanych identyfikatorach.
serviceids	ID/array	Zwraca tylko SLA pasujące do określonych usług.
selectSchedule	query	Zwraca właściwość `schedule` z harmonogramami SLA. Obsługuje `count`.
selectExcludedDowntimes	query	Zwraca właściwość `excluded_downtimes` z wykluczonymi przestojami SLA. Obsługuje `count`.
selectServiceTags	query	Zwraca właściwość `service_tags` z tagami usług SLA. Obsługuje `count`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `slaid`, `name`, `period`, `slo`, `effective_date`, `timezone`, `status`, `description`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został wykorzystany.

Przykłady

Pobieranie wszystkich danych o SLA

Pobieranie danych o wszystkich SLA i ich właściwościach.

Żądanie:

```
{
"jsonrpc": "2.0",
"metoda": "sla.get",
"parametry": {
"wyjście": "rozszerz",
"selectSchedule": ["period_from", "period_to"],
"selectExcludedDowntimes": ["nazwa", "period_from", "period_to"],
"selectServiceTags": ["tag", "operator", "wartość"],
"preservekeys": prawda
},
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"wynik": {
"1": {
"slaid": "1",
"nazwa": "Czas działania bazy danych",
„period”: „1”,

slo”: „99.9995”,

effective_date”: „1672444800”,

timezone”: „America/Toronto”,

status”: „1”,

description”: „Zapewnij doskonały czas sprawności głównych silników baz danych SQL.”,

service_tags”: [
{
„tag”: „database”,

„operator”: „0”,

„value”: „mysql”
},
{
„tag”: „database”,

operator”: „0”,

„value”: „postgresql”
}
],

schedule”: [
{
„period_from”: „0”,

„period_to”: „601200”
}
],
```

```
excluded_downtimes": [
{
„name": „Wdrażanie aktualizacji wersji oprogramowania",
„okres_od": „1648760400",
„okres_do": „1648764900"
}
]
}
},
„id": 1
}
```


Źródło

CSla:get() w *ui/include/classes/api/services/CSla.php*.

sla.getsli

Opis

`object sla.getsli(object parameters)`

Ta metoda umożliwia obliczanie danych Service Level Indicator (SLI) dla Service Level Agreement (SLA).

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object)` Parametry zawierające identyfikator SLA, okresy raportowania oraz opcjonalnie identyfikatory usług, dla których ma zostać obliczony wskaźnik SLI.

Parametr	Typ	Opis
slaid	ID	Identyfikator SLA, dla którego mają zostać zwrócone informacje o dostępności. Zachowanie parametru: - *wymagany*
period_from	timestamp	Początkowy znacznik czasu (włącznie), dla którego ma zostać zaraportowany wskaźnik SLI. Możliwe wartości: znacznik czasu Unix.
period_to	timestamp	Końcowy znacznik czasu (włącznie), dla którego ma zostać zaraportowany wskaźnik SLI. Możliwe wartości: znacznik czasu Unix.
periods	integer	Liczba okresów do zaraportowania. Możliwe wartości: 1-100
serviceids	ID/array	Identyfikatory usług, dla których ma zostać zwrócony wskaźnik SLI.

Podział na okresy

Poniższa tabela przedstawia układ zwracanych wycinków okresów na podstawie kombinacji parametrów.

Note:
Zwracane okresy nie poprzedzają pierwszego dostępnego okresu (na podstawie daty wejścia SLA w życie) i nie wykraczają poza bieżący okres.

Parametry			Zwracane okresy
period_from	**period_to**	**periods**	
-	-	-	Ostatnie 20 okresów, w tym bieżący.
-	-	N	Ostatnie N okresów.
-	specified	-	Ostatnie 20 okresów przed `period_to`.
-	specified	N	Ostatnie N okresów przed `period_to`.
specified	-	-	Pierwsze 20 okresów zaczynając od `period_from`.
specified	-	N	Pierwsze N okresów zaczynając od `period_from`.
specified	specified	-	Do 100 okresów w określonym zakresie.
specified	specified	N	N okresów w określonym zakresie.

Zwracane wartości

`(object)` Zwraca wyniki obliczenia.

Właściwość	Type	Opis
periods	array	Lista raportowanych okresów. Każdy raportowany okres jest reprezentowany jako obiekt składający się z: - `period_from` - `(timestamp)` Początkowy znacznik czasu (włącznie) raportowanego okresu. - `period_to` - `(timestamp)` Końcowy znacznik czasu (wyłącznie) raportowanego okresu. Okresy są sortowane według `period_from`, przy czym najwcześniejsze okresy pojawiają się jako pierwsze.
serviceids	array	Lista identyfikatorów usług w raportowanych okresach. Kolejność sortowania listy nie jest określona. Nawet jeśli parametr `serviceids` został przekazany do metody `sla.getsli`.
sli	array	Dane SLI (jako **tablica dwuwymiarowa**) dla każdego raportowanego okresu i usługi. Indeks właściwości `periods` jest używany jako **pierwszy** wymiar właściwości `sli`. Indeks właściwości `serviceids` jest używany jako **drugi** wymiar właściwości `sli`.

Dane SLI

Dane SLI zwracane dla każdego raportowanego okresu i serwisu obejmują:

Właściwość	Type	Opis
uptime	integer	Czas, przez jaki serwis znajdował się w stanie *OK* podczas zaplanowanego czasu działania, pomniejszony o wykluczone przestoje.
downtime	integer	Czas, przez jaki serwis znajdował się w stanie *not OK* podczas zaplanowanego czasu działania, pomniejszony o wykluczone przestoje.
sli	float	SLI (procent całkowitego czasu działania), na podstawie czasu działania i przestoju.
error_budget	integer	Budżet błędów (w sekundach), na podstawie SLI i SLO.

Właściwość	Type	Opis
excluded_downtimes	array	Tablica wykluczonych przestojów w tym okresie raportowania. Każdy obiekt będzie zawierał następujące parametry: - `name` - (`string`) Nazwa wykluczonego przestoju. - `period_from` - (`timestamp`) Znacznik czasu rozpoczęcia (włącznie) wykluczonego przestoju. - `period_to` - (`timestamp`) Znacznik czasu zakończenia (wyłącznie) wykluczonego przestoju. Wykluczone przestoje są sortowane według `period_from`, przy czym najwcześniejsze okresy są wyświetlane jako pierwsze.

Przykłady

Obliczanie SLI dla dziennego SLA

Pobierz dane SLI dla usług o identyfikatorach „1” i „4”, które są powiązane ze SLA o identyfikatorze „1”. Pobierz dane dla jednego okresu do „1761861599” (30 paź 2025 23:59:59 GMT+0200). Ponieważ okres raportowania SLA jest dzienny, dane SLI są pobierane od „1761775200” (30 paź 2025 00:00:00 GMT+0200) do „1761861600” (31 paź 2025 00:00:00 GMT+0200).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "1",
        "serviceids": [
            1,
            4
        ],
        "periods": 1,
        "period_to": 1761861599
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Obliczanie SLI dla miesięcznego SLA

Pobierz dane SLI o usługach o identyfikatorach „50”, „60” i „70”, które są powiązane ze SLA o identyfikatorze „5”. Pobierz dane dla trzech okresów rozpoczynających się od „1635724800” (01 lis 2021 00:00:00 UTC). Ponieważ okres raportowania SLA jest miesięczny, dane SLI są pobierane dla następujących trzech miesięcy:

- Od „1635724800” (01 lis 2021 00:00:00 UTC) do „1638316800” (01 gru 2021 00:00:00 UTC)
- Od „1638316800” (01 gru 2021 00:00:00 UTC) do „1640995200” (01 sty 2022 00:00:00 UTC)
- Od „1640995200” (01 sty 2022 00:00:00 UTC) do „1643673600” (01 lut 2022 00:00:00 UTC)

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.getsli",
    "params": {
        "slaid": "5",
        "serviceids": [
            50,
            60,
            70
        ],
        "periods": 3,
        "period_from": 1635724800
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": []
                },
                {
                    "uptime": 1674000,
                    "downtime": 0,
                    "sli": 100,
                    "error_budget": 0,
                    "excluded_downtimes": []
                }
            ]
        ]
    },
    "id": 1
}
```


Źródło

CSla::getSli() w *ui/include/classes/api/services/CSla.php*

sla.update

Opis

`object sla.update(object/array slaids)`

Ta metoda umożliwia aktualizację istniejących wpisów SLA.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości SLA do zaktualizowania.

Właściwość `slaid` musi być zdefiniowana dla każdego SLA, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości SLA, metoda akceptuje następujące parametry.

Parametr	Type	Opis
service_tags	array	Tagi usług SLA zastępujące bieżące tagi usług SLA.
schedule	array	Harmonogram SLA zastępujący bieżący. Podanie pustego parametru będzie interpretowane jako harmonogram 24x7.
excluded_downtimes	array	Wykluczone przestoje SLA zastępujące bieżące.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych SLA we właściwości `slaids`.

Przykłady

Aktualizacja tagów usługi

Ustaw SLA o ID „5” tak, aby było obliczane w interwałach miesięcznych dla usług związanych z NoSQL, bez zmiany jego harmonogramu ani wykluczonych przestojów; ustaw SLO na 95%.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
```

```
        ]
    },
    "id": 1
}
```

Zmiana harmonogramu SLA

Przełącz SLA o ID „5” na harmonogram 24x7.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "sla.update",
    "params": {
        "slaid": "5",
        "schedule": []
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
        ]
    },
    "id": 1
}
```

Zmiana wykluczonych przestojów dla SLA

Dodaj planowany 4-godzinny przestój związany z rozbudową pamięci RAM w dniu 6 kwietnia 2022 r., zachowując jednocześnie (należy go zdefiniować ponownie) wcześniej istniejący plan aktualizacji oprogramowania z 4 lipca dla SLA o ID „5”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "slaids": [
            "5"
```

```
        ]
    },
    "id": 1
}
```

Źródło

CSla::update() w *ui/include/classes/api/services/CSla.php*.

Szablon

Ta klasa jest przeznaczona do pracy z szablonami.

Odwołania do obiektów:

- Szablon
- Tag szablonu

Dostępne metody:

- template.create - tworzenie nowych szablonów
- template.delete - usuwanie szablonów
- template.get - pobieranie szablonów
- template.massadd - dodawanie powiązanych obiektów do szablonów
- template.massremove - usuwanie powiązanych obiektów z szablonów
- template.update - aktualizowanie szablonów

Obiekt szablonu

Następujące obiekty są bezpośrednio powiązane z API `template`.

Szablon

Obiekt szablonu ma następujące właściwości.

Właściwość	Typ	Opis
templateid	ID	ID szablonu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	text	Opis szablonu.
host	string	Nazwa techniczna szablonu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Widoczna nazwa szablonu. Domyślnie: wartość właściwości `host`.
readme	text	Instrukcje konfiguracji specyficzne dla szablonu, wyświetlane w Kreatorze hosta. Obsługuje formatowanie Markdown.
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych szablonów z już istniejącymi. Generowany automatycznie, jeśli nie został podany.
vendor_name	string	Nazwa dostawcy szablonu. W przypadku operacji tworzenia zarówno `vendor_name`, jak i `vendor_version` powinny być ustawione albo pozostawione puste. W przypadku operacji aktualizacji `vendor_version` może pozostać puste, jeśli ma wartość w bazie danych.

Właściwość	Typ	Opis
vendor_version	string	Wersja dostawcy szablonu. W przypadku operacji tworzenia zarówno `vendor_name`, jak i `vendor_version` powinny być ustawione albo pozostawione puste. W przypadku operacji aktualizacji `vendor_name` może pozostać puste, jeśli ma wartość w bazie danych.
wizard_ready	integer	Określa, czy szablon jest dostępny do wyboru w Kreatorze hosta. Możliwe wartości: 0 - *(domyślnie)* Niedostępny. 1 - Dostępny.

Znacznik szablonu

Obiekt znacznika szablonu ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa znacznika szablonu. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika szablonu. Zachowanie właściwości: - *tylko do odczytu*.
object	integer	Typ obiektu, z którego znacznik został odziedziczony. Możliwe wartości: 0 - szablon. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego znacznik został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.

template.create

Opis

`object template.create(object/array templates)`

Ta metoda umożliwia tworzenie nowych szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Szablony do utworzenia.

Oprócz standardowych właściwości szablonu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
groups	object/array	Grupy szablonów, do których należy dodać szablon. Grupy szablonów muszą mieć zdefiniowaną wyłącznie właściwość `groupid`. Zachowanie parametrów: - *wymagane*
tags	object/array	Tagi szablonu.
templates	object/array	Szablony, które mają zostać podlinkowane do szablonu. Szablony muszą mieć zdefiniowaną wyłącznie właściwość `templateid`.
macros	object/array	Makra użytkownika, które mają zostać utworzone dla szablonu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych szablonów we właściwości `templateids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych szablonów.

Przykłady

Tworzenie szablonu

Utwórz szablon z tagami i podłącz dwa szablony do tego szablonu.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "11117"
        ]
    },
    "id": 1
}
```

Źródło

CTemplate::create() w *ui/include/classes/api/services/CTemplate.php*.

template.delete

Opis

`object template.delete(array templateIds)`

Ta metoda umożliwia usuwanie szablonów.

Usunięcie szablonu powoduje również usunięcie wszystkich jego encji (pozycji, wyzwalaczy, wykresów itp.) z dowolnych hostów lub szablonów, z którymi jest powiązany. Aby zachować te encje, najpierw odłącz szablon od odpowiednich hostów lub szablonów, używając metod `host.update`, `host.massremove`, `template.update` lub `template.massremove`.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji znajdziesz w sekcji Role użytkowników.

Parametry

`(array)` ID szablonów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych szablonów we właściwości `templateids`.

Przykłady

Usuwanie wielu szablonów

Usuń dwa szablony.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.delete",
    "params": [
        "13",
        "32"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "13",
            "32"
        ]
    },
    "id": 1
}
```

Źródło

CTemplate::delete() w *ui/include/classes/api/services/CTemplate.php*.

template.get

Opis

`integer/array template.get(object parameters)`

Metoda umożliwia pobieranie szablonów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
templateids	ID/array	Zwracaj tylko szablony o podanych ID szablonów.
groupids	ID/array	Zwracaj tylko szablony należące do podanych grup szablonów.
parentTemplateids	ID/array	Zwracaj tylko szablony, do których podany szablon jest podłączony.
hostids	ID/array	Zwracaj tylko szablony, które są podłączone do podanych hostów/szablonów.
graphids	ID/array	Zwracaj tylko szablony zawierające podane wykresy.
itemids	ID/array	Zwracaj tylko szablony zawierające podane pozycje.
triggerids	ID/array	Zwracaj tylko szablony zawierające podane wyzwalacze.
with_items	flag	Zwracaj tylko szablony, które mają pozycje.
with_triggers	flag	Zwracaj tylko szablony, które mają wyzwalacze.
with_graphs	flag	Zwracaj tylko szablony, które mają wykresy.
with_httptests	flag	Zwracaj tylko szablony, które mają scenariusze WWW.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	object/array	Zwracaj tylko szablony z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie szablony. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
inheritedTags	boolean	Zwracaj szablony, które mają podane `tags` również w podłączonych szablonach. Możliwe wartości: `true` - podłączone szablony również muszą mieć podane tagi; `false` - *(domyślnie)* tagi z podłączonych szablonów są ignorowane.
selectInheritedTags	query	Zwracaj właściwość `inheritedTags` z tagami znajdującymi się na podłączonych szablonach.
selectTags	query	Zwracaj tagi szablonu we właściwości `tags`.
selectDiscoveryRules	query	Zwracaj właściwość `discoveryRules` z regułami LLD szablonu. Obsługuje `count`.
selectHosts	query	Zwracaj hosty podłączone do szablonu we właściwości `hosts`. Obsługuje `count`.
selectTemplateGroups	query	Zwracaj grupy szablonów, do których należy szablon, we właściwości `templategroups`.
selectTemplates	query	Zwracaj szablony, do których podany szablon jest podłączony, we właściwości `templates`. Obsługuje `count`.

Parametr	Type	Opis
selectParentTemplates	query	Zwracaj szablony, które są podłączone do podanego szablonu, we właściwości `parentTemplates`. Obsługuje `count`.
selectHttpTests	query	Zwracaj scenariusze WWW z szablonu we właściwości `httpTests`. Obsługuje `count`.
selectItems	query	Zwracaj pozycje z szablonu we właściwości `items`. Obsługuje `count`.
selectTriggers	query	Zwracaj wyzwalacze z szablonu we właściwości `triggers`. Obsługuje `count`.
selectGraphs	query	Zwracaj wykresy z szablonu we właściwości `graphs`. Obsługuje `count`.
selectMacros	query	Zwracaj makra z szablonu we właściwości `macros`.
selectDashboards	query	Zwracaj pulpity z szablonu we właściwości `dashboards`. Obsługuje `count`.
selectValueMaps	query	Zwracaj właściwość `valuemaps` z mapami wartości szablonu.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectTemplates` - wyniki będą sortowane według `name`; `selectHosts` - sortowane według `host`; `selectParentTemplates` - sortowane według `host`; `selectItems` - sortowane według `name`; `selectDiscoveryRules` - sortowane według `name`; `selectTriggers` - sortowane według `description`; `selectGraphs` - sortowane według `name`; `selectDashboards` - sortowane według `name`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `hostid`, `host`, `name`, `status`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
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	Zwracaj mechanizmy wykrywania niskiego poziomu z szablonu we właściwości `discoveries`. Obsługuje `count`. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryRules`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie szablonów według nazwy

Pobierz wszystkie dane o dwóch szablonach o nazwach „Linux by Zabbix agent” i „Windows by Zabbix agent”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent",
                "Windows by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "To jest oficjalny szablon Linux. Wymaga Zabbix agent 8.0 lub nowszego.\r\n\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": "## Przegląd\r\n\r\nTo jest oficjalny szablon Linux. Wymaga Zabbix agent 8.0 lub now
        },
        {
            "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": "To jest oficjalny szablon Windows. Wymaga Zabbix agent 8.0 lub nowszego.\r\n\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": "## Przegląd\r\n\r\nTo jest oficjalny szablon Windows. Wymaga Zabbix agent 8.0 lub n
        }
    ],
    "id": 1
}
```


Pobieranie grup szablonów

Pobierz grupy szablonów, do których należy szablon „Linux by Zabbix agent”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTemplateGroups": "extend",
        "filter": {
            "host": [
                "Linux by Zabbix agent"
            ]
        }
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "templategroups": [
                {
                    "groupid": "10",
                    "name": "Templates/Operating systems",
                    "uuid": "846977d1dfed4968bc5f8bdb363285bc"
                }
            ]
        }
    ],
```

```
    "id": 1
}
```

Pobieranie hostów według szablonu

Pobierz hosty, które mają podłączony szablon „10001” (*Linux by Zabbix agent*).

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "templateid",
        "templateids": "10001",
        "selectHosts": ["hostid", "name"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ],
    "id": 1
}
```

Wyszukiwanie według tagów szablonu

Pobierz szablony, które mają tag „host-name” równy „{HOST.NAME}”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": ["hostid"],
        "selectTags": "extend",
        "evaltype": 0,
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}",
                "operator": 1
            }
        ]
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10402",
            "tags": [
                {
                    "tag": "host-name",
                    "value": "{HOST.NAME}"
                }
            ]
        }
    ],
    "id": 1
}
```

Zobacz także

- Grupa szablonów
- Szablon
- Makro użytkownika
- Interfejs hosta

Źródło

CTemplate::get() w *ui/include/classes/api/services/CTemplate.php*.

template.massadd

Opis

`object template.massadd(object parameters)`

Ta metoda umożliwia jednoczesne dodanie wielu powiązanych obiektów do wskazanych szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory szablonów do zaktualizowania oraz obiekty do dodania do szablonów.

Metoda akceptuje następujące parametry.

Parametr	Type	Opis
templates	object/array	Szablony do zaktualizowania. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`. Zachowanie parametrów: - *wymagane*
groups	object/array	Grupy szablonów do dodania do podanych szablonów. Grupy szablonów muszą mieć zdefiniowaną tylko właściwość `groupid`.
macros	object/array	Makra użytkownika do utworzenia dla podanych szablonów.
templates_link	object/array	Szablony do podlinkowania do podanych szablonów. Szablony muszą mieć zdefiniowaną tylko właściwość `templateid`.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów we właściwości templateids.

Przykłady

Powiąż grupę z szablonami

Dodaj grupę szablonów „2” do dwóch szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10085"
            },
            {
                "templateid": "10086"
            }
        ],
        "groups": [
            {
                "groupid": "2"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Podłącz dwa szablony do szablonu

Podłącz szablony „10106” i „10104” do szablonu „10073”.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "template.massadd",
    "params": {
        "templates": [
            {
                "templateid": "10073"
            }
        ],
        "templates_link": [
            {
                "templateid": "10106"
            },
            {
                "templateid": "10104"
            }
        ]
```

```
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10073"
        ]
    },
    "id": 1
}
```

Zobacz także

- template.update
- Host
- Grupa szablonów
- Makro użytkownika

Źródło

CTemplate::massAdd() w *ui/include/classes/api/services/CTemplate.php*.

template.massremove

Opis

`object template.massremove(object parameters)`

Ta metoda umożliwia usuwanie powiązanych obiektów z wielu szablonów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry zawierające identyfikatory szablonów do zaktualizowania oraz obiekty, które powinny zostać usunięte.

Parametr	Typ	Opis
templateids	ID/array	Identyfikatory szablonów, które mają zostać zaktualizowane. Zachowanie parametru: - *wymagany*
groupids	ID/array	Identyfikatory grup szablonów, z których należy usunąć podane szablony.
macros	string/array	Identyfikatory makr użytkownika, które mają zostać usunięte z podanych szablonów.
templateids_clear	ID/array	Identyfikatory szablonów, które mają zostać odłączone i wyczyszczone z podanych szablonów (nadrzędnych).
templateids_link	ID/array	Identyfikatory szablonów, które mają zostać odłączone od podanych szablonów (nadrzędnych).

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów we właściwości `templateids`.

Przykłady

Usuwanie szablonów z grupy

Usuń dwa szablony z grupy „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": [
            "10085",
            "10086"
        ],
        "groupids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085",
            "10086"
        ]
    },
    "id": 1
}
```

Odłączanie szablonów od hosta

Odłącz szablony „10106” i „10104” od szablonu „10085”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.massremove",
    "params": {
        "templateids": "10085",
        "templateids_link": [
            "10106",
            "10104"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10085"
        ]
    },
    "id": 1
}
```

Zobacz także

- template.update
- Makro użytkownika

Źródło

CTemplate::massRemove() w *ui/include/classes/api/services/CTemplate.php*.

template.update

Opis

`object template.update(object/array templates)`

Ta metoda umożliwia aktualizację istniejących szablonów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości szablonu do zaktualizowania.

Właściwość `templateid` musi być zdefiniowana dla każdego szablonu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko podane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości szablonu metoda akceptuje następujące parametry.

Parametr	Typ	Opis
groups	object/array	Grupy szablonów zastępujące bieżące grupy szablonów, do których należą szablony. Dla grup szablonów musi być zdefiniowana tylko właściwość `groupid`.
tags	object/array	Tagi szablonu zastępujące bieżące tagi szablonu.
macros	object/array	Makra użytkownika zastępujące bieżące makra użytkownika w podanych szablonach.
templates	object/array	Szablony zastępujące aktualnie podlinkowane szablony. Szablony, które nie zostaną przekazane, zostaną jedynie odlinkowane. Dla szablonów musi być zdefiniowana tylko właściwość `templateid`.
templates_clear	object/array	Szablony do odlinkowania i wyczyszczenia z podanych szablonów. Dla szablonów musi być zdefiniowana tylko właściwość `templateid`.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych szablonów we właściwości `templateids`.

Przykłady

Zmiana standardowych właściwości szablonu

Zmień nazwę techniczną szablonu na „Linux by Zabbix agent Custom”, nazwę widoczną na „My template” oraz zaktualizuj opis szablonu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "host": "Linux by Zabbix agent Custom",
        "name": "My template",
        "description": "This is a custom Linux template."
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
```

```
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie grup szablonów

Zastąp wszystkie grupy szablonów dla danego szablonu inną grupą.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "groups": [
            {
                "groupid": "24"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie wielu grup szablonów

Zastąp wszystkie grupy szablonów dla wielu szablonów innymi grupami.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086",
            "10087",
            "10088"
        ]
    },
    "id": 1
}
```

Aktualizowanie tagów szablonu

Zastąp wszystkie tagi szablonu innym tagiem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "tags": [
            {
                "tag": "host-name",
                "value": "{HOST.NAME}"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie makr szablonu

Zastąp wszystkie makra szablonu innymi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "macros": [
            {
                "macro": "{$MY_MACRO}",
                "value": "new_value"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Aktualizowanie wielu makr szablonu

Zastąp wszystkie makra użytkownika podanym makrem użytkownika w wielu szablonach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": [
            {
                "templateid": "10087",
                "macros": [
                    {
                        "macro": "{$AGENT.TIMEOUT}",
                        "value": "5m",
                        "description": "Limit czasu, po którym agent jest uznawany za niedostępny. Działa
                    }
                ]
            },
            {
                "templateid": "10088",
                "macros": [
                    {
                        "macro": "{$AGENT.TIMEOUT}",
                        "value": "5m",
                        "description": "Limit czasu, po którym agent jest uznawany za niedostępny. Działa
                    }
                ]
            }
        ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
```

```
    "result": {
        "templateids": [
            "10087",
            "10088"
        ]
    },
    "id": 1
}
```

Aktualizacja podłączonych szablonów szablonu

Odłącz (bez czyszczenia) wszystkie szablony od podanego szablonu i podłącz do niego inny.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": {
        "templateid": "10086",
        "templates": [
            {
                "templateid": "10001"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10086"
        ]
    },
    "id": 1
}
```

Czyszczenie wielu podłączonych szablonów szablonu

Odłącz i wyczyść określony podłączony szablon z podanych szablonów.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "template.update",
    "params": [
        {
            "templateid": "10087",
            "templates_clear": [
                {
                    "templateid": "10001"
                }
            ]
        },
        {
            "templateid": "10088",
            "templates_clear": [
                {
                    "templateid": "10001"
                }
            ]
        }
```

```
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "templateids": [
            "10087",
            "10088"
        ]
    },
    "id": 1
}
```

Źródło

CTemplate::update() w *ui/include/classes/api/services/CTemplate.php*.

Token

Ta klasa jest przeznaczona do pracy z tokenami.

Odwołania do obiektów:

- Token

Dostępne metody:

- token.create - tworzenie nowych tokenów
- token.delete - usuwanie tokenów
- token.get - pobieranie tokenów
- token.update - aktualizowanie tokenów
- token.generate - generowanie tokenów

Obiekt tokena

Następujące obiekty są bezpośrednio powiązane z API `token`.

Token

Obiekt tokenu ma następujące właściwości.

Właściwość	Typ	Opis
tokenid	ID	ID tokenu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
name	string	Nazwa tokenu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
description	text	Opis tokenu.
userid	ID	ID użytkownika, do którego przypisano token. Domyślnie: *bieżący użytkownik*. Zachowanie właściwości: - *stałe*

Właściwość	Typ	Opis
lastaccess	timestamp	Data i godzina ostatniego uwierzytelnienia tokenu. ”0”, jeśli token nigdy nie został uwierzytelniony. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Status tokenu. Możliwe wartości: 0 - *(domyślnie)* token włączony; 1 - token wyłączony.
expires_at	timestamp	Data i godzina wygaśnięcia tokenu. ”0” dla tokenów, które nigdy nie wygasają.
created_at	timestamp	Data i godzina utworzenia tokenu. Zachowanie właściwości: - *tylko do odczytu*
creator_userid	ID	ID użytkownika, który utworzył token. Zachowanie właściwości: - *tylko do odczytu*

token.create

Opis

`object token.create(object/array tokens)`

Ta metoda umożliwia tworzenie nowych tokenów.

Note:
Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Attention:
Token utworzony tą metodą musi również zostać wygenerowany, zanim będzie można go używać.

Parametry

`(object/array)` Tokeny do utworzenia.

Metoda akceptuje tokeny ze standardowymi właściwościami tokenu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych tokenów we właściwości `tokenids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych tokenów.

Przykłady

Utwórz token

Utwórz włączony token, który nigdy nie wygasa i uwierzytelnia użytkownika o ID 2.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Your token",
        "userid": "2"
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "188"
        ]
    },
    "id": 1
}
```

Utwórz wyłączony token, który wygasa 21 stycznia 2021 roku. Ten token będzie uwierzytelniał bieżącego użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.create",
    "params": {
        "name": "Your token",
        "status": "1",
        "expires_at": "1611238072"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "189"
        ]
    },
    "id": 1
}
```

Źródło

CToken::create() w *ui/include/classes/api/services/CToken.php*.

token.delete

Opis

`object token.delete(array tokenids)`

Ta metoda umożliwia usuwanie tokenów.

> **Note:**
> Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Parametry

`(array)` ID tokenów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych tokenów we właściwości `tokenids`.

Przykłady

Usuń wiele tokenów

Usuń dwa tokeny.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "token.delete",
"params": [
"188",
"192"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"tokenids": [
"188",
"192"
]
},
"id": 1
}
```

Źródło

CToken::delete() w *ui/include/classes/api/services/CToken.php*.

token.generate

Opis

`object token.generate(array tokenids)`

Ta metoda umożliwia generowanie tokenów.

Note:
Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Attention:
Token może zostać wygenerowany tą metodą tylko wtedy, gdy został utworzony.

Parametry

`(array)` ID tokenów do wygenerowania.

Zwracane wartości

`(array)` Zwraca tablicę obiektów zawierających identyfikator wygenerowanego tokenu we właściwości `tokenid` oraz wygenerowany ciąg autoryzacyjny we właściwości `token`.

Właściwość	Type	Opis
tokenid	ID	Identyfikator tokenu.
token	string	Wygenerowany ciąg autoryzacyjny dla tego tokenu.

Przykłady

Wygeneruj wiele tokenów

Wygeneruj dwa tokeny.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.generate",
    "params": [
        "1",
        "2"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "tokenid": "1",
            "token": "bbcfce79a2d95037502f7e9a534906d3466c9a1484beb6ea0f4e7be28e8b8ce2"
        },
        {
            "tokenid": "2",
            "token": "fa1258a83d518eabd87698a96bd7f07e5a6ae8aeb8463cae33d50b91dd21bd6d"
        }
    ],
    "id": 1
}
```

Źródło

CToken::generate() w *ui/include/classes/api/services/CToken.php*.

token.get

Opis

`integer/array token.get(object parameters)`

Metoda umożliwia pobieranie tokenów zgodnie z podanymi parametrami.

Note:
Tylko użytkownik typu *Super admin* ma uprawnienia do wyświetlania tokenów innych użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
tokenids	ID/array	Zwracaj tylko tokeny o podanych identyfikatorach.
userids	ID/array	Zwracaj tylko tokeny utworzone dla podanych użytkowników.
token	string	Zwracaj tylko tokeny utworzone dla podanego *tokenu uwierzytelniającego*.
valid_at	timestamp	Zwracaj tylko tokeny, które są ważne (nie wygasły) w podanej dacie i czasie.
expired_at	timestamp	Zwracaj tylko tokeny, które są wygasłe (nieważne) w podanej dacie i czasie.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `tokenid`, `name`, `lastaccess`, `status`, `expires_at`, `created_at`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	

Parametr	Typ	Opis
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie tokena

Pobierz wszystkie dane tokena o ID „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.get",
    "params": {
        "output": "extend",
        "tokenids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CToken::get() w *ui/include/classes/api/services/CToken.php*.

token.update

Opis

`object token.update(object/array tokens)`

Ta metoda umożliwia aktualizację istniejących tokenów.

Note:
Uprawnienie *Manage API tokens* permission jest wymagane, aby rola użytkownika mogła zarządzać tokenami innych użytkowników.

Parametry

`(object/array)` Właściwości tokenów do zaktualizowania.

Właściwość `tokenid` musi być zdefiniowana dla każdego tokenu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Metoda akceptuje tokeny ze standardowymi właściwościami tokenu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych tokenów we właściwości `tokenids`.

Przykłady

Usuń datę wygaśnięcia tokena

Usuń datę wygaśnięcia z tokena.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "token.update",
    "params": {
        "tokenid": "2",
        "expires_at": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "tokenids": [
            "2"
        ]
    },
    "id": 1
}
```

Źródło

CToken::update() w *ui/include/classes/api/services/CToken.php*.

Trend

Ta klasa została zaprojektowana do pracy z danymi trendów.

Odwołania do obiektów:

- Trend zmiennoprzecinkowy
- Trend całkowitoliczbowy

Dostępne metody:

- trend.get — pobieranie trendów

Obiekt trendu

Następujące obiekty są bezpośrednio powiązane z API `trend`.

Note:
Obiekty trendów różnią się w zależności od typu informacji elementu. Są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pośrednictwem API.

Trend float

Obiekt trendu float ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Znacznik czasu godziny, dla której obliczono wartość. Na przykład znacznik czasu „04:00:00” oznacza wartości obliczone dla okresu „04:00:00-04:59:59”.
itemid	ID	ID powiązanej pozycji.
num	integer	Liczba wartości, które były dostępne dla danej godziny.
value_min	float	Minimalna wartość godzinowa.
value_avg	float	Średnia wartość godzinowa.
value_max	float	Maksymalna wartość godzinowa.

Trend całkowitoliczbowy

Obiekt trendu całkowitoliczbowego ma następujące właściwości.

Właściwość	Typ	Opis
clock	timestamp	Znacznik czasu godziny, dla której obliczono wartość. Na przykład znacznik czasu „04:00:00” oznacza wartości obliczone dla okresu „04:00:00-04:59:59”.
itemid	ID	ID powiązanej pozycji.
num	integer	Liczba wartości, które były dostępne dla danej godziny.
value_min	integer	Minimalna wartość godzinowa.
value_avg	integer	Średnia wartość godzinowa.
value_max	integer	Maksymalna wartość godzinowa.

trend.get

Opis

`integer/array trend.get(object parameters)`

Metoda umożliwia pobieranie danych trendów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Type	Opis
itemids	ID/array	Zwracaj tylko trendy dla podanych identyfikatorów pozycji.
time_from	timestamp	Zwracaj tylko wartości, które zostały zebrane po podanym czasie lub o podanym czasie.
time_till	timestamp	Zwracaj tylko wartości, które zostały zebrane przed podanym czasem lub o podanym czasie.
countOutput	boolean	Policz liczbę pobranych obiektów.
limit	integer	Ogranicz liczbę pobranych obiektów.
output	query	Ustaw właściwości obiektu trendu, które mają zostać zwrócone.

Zwracane wartości

(integer/array) Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych trendów pozycji

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trend.get",
    "params": {
        "output": [
            "itemid",
            "clock",
            "num",
            "value_min",
            "value_avg",
            "value_max"
        ],
        "itemids": [
            "23715"
        ],
        "limit": "1"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "itemid": "23715",
            "clock": "1446199200",
            "num": "60",
            "value_min": "0.165",
            "value_avg": "0.2168",
            "value_max": "0.35"
        }
    ],
    "id": 1
}
```

Źródło

CTrend::get() w *ui/include/classes/api/services/CTrend.php*.

Typ nośnika

Ta klasa jest przeznaczona do pracy z typami nośników.

Odwołania do obiektów:

- Typ nośnika
  - Parametry webhooka
  - Parametry skryptu
- Szablon wiadomości

Dostępne metody:

- mediatype.create - utwórz nowe typy nośników
- mediatype.delete - usuń typy nośników
- mediatype.get - pobierz typy nośników
- mediatype.update - zaktualizuj typy nośników

Obiekt typu mediów

Następujące obiekty są bezpośrednio powiązane z API `mediatype`.

Typ nośnika

Obiekt typu nośnika ma następujące właściwości.

Właściwość	Typ	Opis
mediatypeid	ID	ID typu nośnika. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* w operacjach aktualizacji
name	string	Nazwa typu nośnika. Zachowanie właściwości: - *wymagana* w operacjach tworzenia
type	integer	Transport używany przez typ nośnika. Możliwe wartości: 0 - Email; 1 - Script; 2 - SMS; 4 - Webhook; 5 - Push.
exec_path	string	Zachowanie właściwości: - *wymagana* w operacjach tworzenia Nazwa pliku skryptu (np. notification.sh) znajdującego się w katalogu określonym w parametrze konfiguracji serwera AlertScriptsPath.
gsm_modem	string	Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Script” Nazwa urządzenia szeregowego modemu GSM.
passwd	string	Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „SMS” Hasło uwierzytelniania.
provider	integer	Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_authentication` ma wartość „Normal password” Dostawca poczty e-mail. Możliwe wartości: 0 - *(domyślnie)* Generic SMTP; 1 - Gmail; 2 - Gmail relay; 3 - Office365; 4 - Office365 relay.
smtp_email	string	Adres e-mail, z którego będą wysyłane powiadomienia. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Email”
smtp_helo	string	SMTP HELO. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Email”
smtp_server	string	Serwer SMTP. Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Email”

Właściwość	Typ	Opis
smtp_port	integer	Port serwera SMTP, z którym należy się połączyć. Wartość domyślna: 25. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Email”
smtp_security	integer	Poziom zabezpieczeń połączenia SMTP, który ma być używany. Możliwe wartości: 0 - *(domyślnie)* None; 1 - STARTTLS; 2 - SSL/TLS. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Email”
smtp_verify_host	integer	Weryfikacja hosta SSL dla SMTP. Możliwe wartości: 0 - *(domyślnie)* No; 1 - Yes. Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_security` ma wartość „STARTTLS” lub „SSL/TLS”
smtp_verify_peer	integer	Weryfikacja peer SSL dla SMTP. Możliwe wartości: 0 - *(domyślnie)* No; 1 - Yes. Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_security` ma wartość „STARTTLS” lub „SSL/TLS”
smtp_authentication	integer	Metoda uwierzytelniania SMTP, która ma być używana. Możliwe wartości: 0 - *(domyślnie)* None; 1 - Normal password; 2 - OAuth token. Uwierzytelnianie OAuth nie jest dozwolone dla dostawcy poczty e-mail *Office365 relay*. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Email”
redirection_url	string	Adres URL frontend Zabbix, do którego nastąpi przekierowanie po autoryzacji OAuth. Wartość domyślna: Wartość właściwości `url` ustawień API z częścią `zabbix.php?action=oauth.authorize` Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”
client_id	string	Identyfikator klienta zarejestrowany na serwerze autoryzacji OAuth. Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”
client_secret	string	Sekret klienta zarejestrowany na serwerze autoryzacji OAuth. Dostępny tylko dla użytkownika typu Super Admin. Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”

Właściwość	Typ	Opis
authorization_url	string	Adres URL OAuth z parametrami służącymi do uzyskania tokenów dostępu i odświeżania. Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”
token_url	string	Adres URL OAuth służący do wymiany tokenu autoryzacji na tokeny dostępu i odświeżania. Ten adres URL jest również używany przez serwer do odświeżania nieprawidłowego tokenu dostępu. Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”
tokens_status	integer	Maska bitowa stanu tokenów. Możliwe wartości: 0 - (domyślnie) Oba tokeny zawierają nieprawidłową wartość 1 - Token dostępu zawiera prawidłową wartość 2 - Token odświeżania zawiera prawidłową wartość 3 - Oba tokeny zawierają prawidłową wartość. Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_authentication` ma wartość „OAuth token”
access_token	string	Wartość tokenu dostępu OAuth. Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”
access_token_updated	timestamp	Znacznik czasu ostatniej modyfikacji `access_token` wykonanej przez serwer podczas odświeżania za pomocą `refresh_token` lub przez API podczas zmian tokenu. Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_authentication` ma wartość „OAuth token”
access_expires_in	integer	Czas w sekundach, po którym `access_token` stanie się nieaktualny i konieczne będzie wysłanie żądania do `refresh_url`. Jest ustawiana przez serwer Zabbix podczas odświeżania `access_token` lub przez API podczas zmian tokenu. Znacznik czasu jest obliczany przez dodanie wartości `access_token_updated`. Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_authentication` ma wartość „OAuth token”
refresh_token	string	Wartość tokenu odświeżania OAuth. Zachowanie właściwości: - *wymagana*, jeśli `smtp_authentication` ma wartość „OAuth token”
status	integer	Określa, czy typ nośnika jest włączony. Możliwe wartości: 0 - *(domyślnie)* Włączony; 1 - Wyłączony.
username	string	Nazwa użytkownika. Zachowanie właściwości: - *obsługiwana*, jeśli `smtp_authentication` ma wartość „Normal password”

Właściwość	Typ	Opis
maxsessions	integer	Maksymalna liczba alertów, które mogą być przetwarzane równolegle. Możliwe wartości, jeśli `type` ma wartość „SMS”: 1. Możliwe wartości, jeśli `type` ma wartość „Email”, „Script” lub „Webhook”: 0-100. Wartość domyślna: 1.
maxattempts	integer	Maksymalna liczba prób wysłania alertu. Możliwe wartości: 1-100. Wartość domyślna: 3.
attempt_interval	string	Odstęp między próbami ponowienia. Akceptuje liczbę sekund (np. 1, 30, 60) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `1h`). Możliwe wartości: 0-1h. Wartość domyślna: 10s.
message_format	integer	Format wiadomości. Możliwe wartości: 0 - Zwykły tekst; 1 - *(domyślnie)* HTML. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Email”
script	text	Treść skryptu webhooka (JavaScript). Zachowanie właściwości: - *wymagana*, jeśli `type` ma wartość „Webhook”
timeout	string	Limit czasu skryptu webhooka. Akceptuje liczbę sekund (np. 1, 60) lub wartość z przyrostkiem czasu (np. `30s`, `1m`) Możliwe wartości: 1-60s. Wartość domyślna: 30s. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Webhook”
process_tags	integer	Przetwarzanie wartości właściwości JSON w odpowiedzi skryptu webhooka jako tagów. Tagi te są dodawane do istniejących tagów problemu. Możliwe wartości: 0 - *(domyślnie)* Ignoruj odpowiedź skryptu webhooka; 1 - Przetwarzaj odpowiedź skryptu webhooka jako tagi. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Webhook”

Właściwość	Typ	Opis
show_event_menu	integer	Uwzględnij wpis w menu zdarzeń zawierający odsyłacz do niestandardowego adresu URL. Dodaje również właściwość `urls` do danych wyjściowych metod problem.get i event.get. Możliwe wartości: 0 - *(domyślnie)* Nie uwzględniaj wpisu w menu zdarzeń ani właściwości `urls`; 1 - Uwzględnij wpis w menu zdarzeń i właściwość `urls`. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Webhook”
event_menu_url	string	Adres URL używany we wpisie menu zdarzeń oraz we właściwości `urls` zwracanej przez metody problem.get i event.get. Zachowanie właściwości: - *obsługiwana*, jeśli `show_event_menu` ma wartość „Include event menu entry and `urls` property”
event_menu_name	string	Nazwa używana we wpisie menu zdarzeń oraz we właściwości `urls` zwracanej przez metody problem.get i event.get. Zachowanie właściwości: - *obsługiwana*, jeśli `show_event_menu` ma wartość „Include event menu entry and `urls` property”
parameters	array	Parametry webhooka lub skryptu. Zachowanie właściwości: - *obsługiwana*, jeśli `type` ma wartość „Webhook” lub „Script”
description	text	Opis typu nośnika.

Parametry webhooka

Parametry webhooka mają następujące właściwości.

Właściwość	Type	Opis
name	string	Nazwa parametru. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru, obsługuje makra. Obsługiwane makra są opisane na stronie Supported macros.

Parametry skryptu

Parametry skryptu mają następujące właściwości.

Właściwość	Type	Opis
sortorder	integer	Kolejność, w jakiej wartości parametrów będą przekazywane do skryptu jako argumenty wiersza poleceń, zaczynając od 0 jako pierwszego argumentu. Zachowanie właściwości: - *wymagane*
value	string	Wartość parametru, obsługuje makra. Obsługiwane makra są opisane na stronie Supported macros.

Szablon wiadomości

Obiekt szablonu wiadomości definiuje szablon, który będzie używany jako domyślna wiadomość dla operacji akcji wysyłających powiadomienie. Ma on następujące właściwości.

Właściwość	Type	Opis
eventsource	integer	Źródło zdarzenia. Możliwe wartości: 0 - Wyzwalacze; 1 - Wykrywanie; 2 - Autorejestracja; 3 - Wewnętrzne; 4 - Usługi. Property behavior: - *required*
recovery	integer	Tryb operacji. Możliwe wartości: 0 - Operacje; 1 - Operacje odzyskiwania; 2 - Operacje aktualizacji. Property behavior: - *required*
subject	string	Temat wiadomości.
message	string	Treść wiadomości.

mediatype.create

Opis

`object mediatype.create(object/array mediaTypes)`

Ta metoda umożliwia utworzenie nowych typów mediów.

Note:
Ta metoda jest dostępna wyłącznie dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli metody dostarczania zostaną określone we fladze funkcji `$ZBX_FEATURE_FLAGS['media_type_denylist']` w pliku konfiguracyjnym frontendu, użytkownik nie może wywołać tej metody dla odpowiednich typów mediów.

Parametry

`(object/array)` Typy mediów do utworzenia.

Oprócz standardowych właściwości typu mediów metoda akceptuje następujące parametry.

Parametr	Type	Opis
message_templates	array	Szablony wiadomości, które mają zostać utworzone dla typu mediów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych typów mediów we właściwości `mediatypeids`. Kolejność zwracanych identyfikatorów odpowiada kolejności przekazanych typów mediów.

Przykłady

Tworzenie typu nośnika e-mail

Utwórz nowy typ nośnika e-mail z niestandardowym portem SMTP i szablonami wiadomości.

Żądanie:

```
{
    "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": "Problem: {EVENT.NAME}",
                "message": "Problem \"{EVENT.NAME}\" na hoście \"{HOST.NAME}\" rozpoczął się o {EVENT.TIME
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "Rozwiązano w {EVENT.DURATION}: {EVENT.NAME}",
                "message": "Problem \"{EVENT.NAME}\" na hoście \"{HOST.NAME}\" został rozwiązany o {EVENT.
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "Zaktualizowany problem w {EVENT.AGE}: {EVENT.NAME}",
                "message": "{USER.FULLNAME} wykonał(a) akcję {EVENT.UPDATE.ACTION} dla problemu \"{EVENT.N
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "7"
        ]
    },
    "id": 1
}
```

Tworzenie typu nośnika skryptu

Utwórz nowy typ nośnika skryptu z niestandardową wartością liczby prób oraz interwału między nimi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.create",
    "params": {
        "type": "1",
        "name": "Push notifications",
        "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "8"
        ]
    },
    "id": 1
}
```

Tworzenie typu nośnika webhook

Utwórz nowy typ nośnika webhook.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "9"
        ]
```

```
    },
    "id": 1
}
```

Źródło

CMediaType::create() w *ui/include/classes/api/services/CMediaType.php*.

mediatype.delete

Opis

`object mediatype.delete(array mediaTypeIds)`

Ta metoda umożliwia usuwanie typów mediów.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli metody dostarczania są określone we fladze funkcji `$ZBX_FEATURE_FLAGS['media_type_denylist']` w pliku konfiguracyjnym frontendu, użytkownik nie może wywołać tej metody dla odpowiednich typów mediów.

Parametry

`(array)` ID typów mediów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych typów mediów we właściwości `mediatypeids`.

Przykłady

Usuwanie wielu typów mediów

Usuń dwa typy mediów.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "mediatype.delete",
"params": [
"3",
"5"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"mediatypeids": [
"3",
"5"
]
},
"id": 1
}
```

Źródło

CMediaType::delete() w *ui/include/classes/api/services/CMediaType.php*.

mediatype.get

Opis

`integer/array mediatype.get(object parameters)`

Metoda umożliwia pobieranie typów nośników zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli metody dostarczania są określone we fladze funkcji `$ZBX_FEATURE_FLAGS['media_type_denylist']` w pliku konfiguracyjnym frontend, użytkownik nie może wywoływać tej metody dla odpowiednich typów nośników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

> **Note:**
> Podczas żądania informacji o typach mediów powiązanych z użytkownikiem, użytkownicy typu *Admin* i *User* mogą pobierać dane tylko o własnym użytkowniku. Przykład znajduje się w sekcji Pobieranie typów mediów jako *Admin*.

Parametr	Typ	Opis
mediatypeids	ID/array	Zwracaj tylko typy mediów o podanych ID.
mediaids	ID/array	Zwracaj tylko typy mediów używane przez podane media.
userids	ID/array	Zwracaj tylko typy mediów używane przez podanych użytkowników.
selectActions	query	Zwracaj właściwość `actions` z akcjami używającymi typu mediów.
selectMessageTemplates	query	Zwracaj właściwość `message_templates` z tablicą komunikatów typu mediów. Zachowanie parametru: - *obsługiwane* dla użytkowników typu *Super admin*
selectUsers	query	Zwracaj właściwość `users` z użytkownikami używającymi typu mediów. Ograniczenia zależne od typu użytkownika opisano w `user.get`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `mediatypeid`.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type.
output	query	Możliwe właściwości obiektu typu mediów dla użytkowników typu *Admin* i *User*: `mediatypeid`, `name`, `type`, `status`, `maxattempts`. Właściwości obiektu typu mediów, które mają zostać zwrócone. Użytkownicy typu *Admin* i *User* mogą pobierać tylko następujące właściwości: `mediatypeid`, `name`, `type`, `status`, `maxattempts`, `description`. Przykład znajduje się w sekcji Pobieranie typów mediów jako *Admin*. Domyślnie: `extend`.

Parametr	Typ	Opis
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Możliwe właściwości obiektu typu mediów dla użytkowników typu *Admin* i *User*: `name`, `description`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie typów mediów

Pobierz wszystkie skonfigurowane typy mediów. Poniższy przykład zwraca dwa typy mediów:

- typ mediów e-mail;
- typ mediów SMS.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "selectMessageTemplates": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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 rozpoczął się o {EVENT.TIME} dnia {EVENT.DATE}\r\nNazwa problemu:
            },
            {
                "eventsource": "0",
                "recovery": "1",
                "subject": "Rozwiązano: {EVENT.NAME}",
                "message": "Problem został rozwiązany o {EVENT.RECOVERY.TIME} dnia {EVENT.RECOVERY.DAT
            },
            {
                "eventsource": "0",
                "recovery": "2",
                "subject": "Zaktualizowany problem: {EVENT.NAME}",
                "message": "{USER.FULLNAME} wykonał(a) akcję {EVENT.UPDATE.ACTION} dla problemu dnia {
            },
            {
                "eventsource": "1",
                "recovery": "0",
                "subject": "Wykrywanie: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}",
                "message": "Reguła wykrywania: {DISCOVERY.RULE.NAME}\r\n\r\nIP urządzenia: {DISCOVERY.
            },
            {
                "eventsource": "2",
                "recovery": "0",
                "subject": "Autorejestracja: {HOST.HOST}",
                "message": "Nazwa hosta: {HOST.HOST}\r\nIP hosta: {HOST.IP}\r\nPort agenta: {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": "ROZWIĄZANO: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT.TIME}
                },
                {
                    "eventsource": "0",
                    "recovery": "2",
                    "subject": "",
                    "message": "{USER.FULLNAME} wykonał(a) akcję {EVENT.UPDATE.ACTION} dla problemu dnia {
                },
                {
                    "eventsource": "1",
                    "recovery": "0",
                    "subject": "",
                    "message": "Wykrywanie: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}"
                },
                {
                    "eventsource": "2",
                    "recovery": "0",
                    "subject": "",
                    "message": "Autorejestracja: {HOST.HOST}\r\nIP hosta: {HOST.IP}\r\nPort agenta: {HOST.
                }
            ],
            "parameters": []
        }
    ],
    "id": 1
}
```


Pobieranie typów mediów jako *Admin*

Jako użytkownik typu *Admin* pobierz wszystkie typy mediów, które są włączone, wraz z użytkownikami korzystającymi z tych typów mediów. Poniższy przykład zwraca dwa typy mediów:

- typ mediów e-mail z jednym użytkownikiem (tylko własny użytkownik użytkownika typu *Admin*);
- typ mediów SMS bez użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": "extend",
        "filter": {
            "status": 0
        },
        "selectUsers": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobieranie typów nośników skryptu i webhooka

Poniższy przykład zwraca trzy typy nośników:

- typ nośnika skryptu z parametrami;
- typ nośnika skryptu bez parametrów;
- typ nośnika webhooka z parametrami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.get",
    "params": {
        "output": ["mediatypeid", "name", "parameters"],
        "filter": {
            "type": [1, 4]
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Użytkownik

Źródło

CMediaType::get() w *ui/include/classes/api/services/CMediaType.php*.

mediatype.update

Opis

`object mediatype.update(object/array mediaTypes)`

Ta metoda umożliwia aktualizowanie istniejących typów nośników.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.
 Jeśli metody dostarczania są określone we fladze funkcji `$ZBX_FEATURE_FLAGS['media_type_denylist']` w pliku konfiguracyjnym frontendu, użytkownik nie może wywoływać tej metody dla odpowiednich typów nośników.

Parametry

`(object/array)` Właściwości typu mediów do zaktualizowania.

Właściwość `mediatypeid` musi być zdefiniowana dla każdego typu mediów, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości typu mediów, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
message_templates	array	Szablony wiadomości zastępujące bieżące szablony wiadomości.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych typów mediów we właściwości `mediatypeids`.

Przykłady

Włączanie typu mediów

Włącz typ mediów, czyli ustaw jego status na ”0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "mediatype.update",
    "params": {
        "mediatypeid": "6",
        "status": "0"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "6"
        ]
    },
    "id": 1
}
```

Źródło

CMediaType::update() w *ui/include/classes/api/services/CMediaType.php*.

Urządzenie

Ta klasa jest przeznaczona do pracy z urządzeniami mobilnymi używanymi z aplikacją mobilną Zabbix.

Odwołania do obiektów:

- Device

Dostępne metody:

- device.get - pobieranie urządzeń
- device.offboard - usuwanie urządzeń (wycofywanie ich z użycia)

Obiekt urządzenia

Obiekt urządzenia reprezentuje urządzenie mobilne używane z aplikacją mobilną Zabbix.

Property	Type	Description
deviceid	ID	Identyfikator urządzenia (wewnętrzny, np. 1, 2). Zachowanie właściwości: - Tylko do odczytu
userid	ID	Identyfikator użytkownika urządzenia. Zachowanie właściwości: - Stała
uuid	string	Uniwersalny unikatowy identyfikator urządzenia (UUIDv7) (zewnętrzny, np. `019f930e-0ea2-7c1f-9dcc-0fbb60c40664`). Zachowanie właściwości: - Tylko do odczytu
name	string	Nazwa urządzenia. Zachowanie właściwości: - Tylko do odczytu

Property	Type	Description
status	integer	Stan urządzenia. Możliwe wartości: 0 - Nowe 1 - Aktywne 2 - Nieprzypisane Zachowanie właściwości: - Tylko do odczytu
activated_at	timestamp	Data i godzina dodania urządzenia w aplikacji mobilnej lub 0, jeśli nie zostało jeszcze dodane. Zachowanie właściwości: - Tylko do odczytu
lastaccess	timestamp	Data i godzina ostatniego uwierzytelnienia urządzenia w aplikacji mobilnej lub 0, jeśli nie zostało jeszcze uwierzytelnione. Zachowanie właściwości: - Tylko do odczytu

device.get

Ta metoda umożliwia pobieranie urządzeń mobilnych używanych z aplikacją mobilną Zabbix.

Metoda jest dostępna dla użytkowników typu *User*, *Admin* i *Super admin*, jeśli parametr `EnableMobileDevices` jest włączony, a użytkownik ma dostęp do urządzeń. Dostęp do metod API można odebrać w konfiguracji roli użytkownika.

Użytkownicy typu *User* i *Admin* mogą pobierać tylko własne urządzenia.

Sygnatura metody: `integer/array device.get(object parameters)`

Źródło metody: `CDevice::get()` w `ui/include/classes/api/services/CDevice.php`

Parametry

Parametr	Typ	Opis
deviceids	ID/array	Zwraca tylko urządzenia o podanych identyfikatorach urządzeń.
userids	ID/array	Zwraca tylko urządzenia należące do użytkowników o podanych identyfikatorach użytkowników.
roleids	ID/array	Zwraca tylko urządzenia należące do użytkowników o podanych identyfikatorach ról użytkowników.
usrgrpids	ID/array	Zwraca tylko urządzenia należące do użytkowników o podanych identyfikatorach grup użytkowników.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `deviceid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Wartości zwracane

Zwraca tablicę obiektów albo liczbę pobranych obiektów (jeśli użyto `countOutput`).

Przykłady

Pobieranie urządzeń

Pobierz urządzenia należące do użytkowników o następujących identyfikatorach użytkowników: 1, 2, 3.
 Należy pamiętać, że użytkownik 2 (gość) nie ma żadnych urządzeń, a urządzenie 3 nie zostało jeszcze dodane ani uwierzytelnione w aplikacji mobilnej Zabbix.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "device.get",
    "params": {
        "output": "extend",
        "userids": [1, 2, 3]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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

Ta metoda umożliwia usunięcie (wyrejestrowanie) urządzeń mobilnych używanych z aplikacją mobilną Zabbix.

Metoda jest dostępna dla użytkowników typu *User*, *Admin* i *Super admin*, jeśli parametr `EnableMobileDevices` jest włączony, a użytkownik ma dostęp do urządzeń. Dostęp do metod API można odebrać w konfiguracji roli użytkownika.

Użytkownicy typu *User* i *Admin* mogą usuwać tylko własne urządzenia.

Sygnatura metody: `object device.offboard(object parameters)`

Źródło metody: `CDevice::offboard()` w pliku `ui/include/classes/api/services/CDevice.php`

Parametry

Parametr	Typ	Opis
uuid	string	Uniwersalny unikatowy identyfikator urządzenia (UUIDv7). Odpowiada `deviceid`.
force	boolean	Ustaw `true`, aby zignorować błąd „Nie znaleziono urządzenia” i wymusić usunięcie urządzenia. Wartość domyślna: `false`

Wartości zwracane

Zwraca obiekt zawierający UUID usuniętego urządzenia.

Przykłady

Usuń urządzenie

Usuwa urządzenie o podanym identyfikatorze UUID, ignorując błąd „Nie znaleziono urządzenia”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "device.offboard",
    "params": {
        "uuid": "019d0106-5eef-7154-a159-bb29e35d38e5",
        "force": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "uuid": "019d0106-5eef-7154-a159-bb29e35d38e5"
    },
    "id": 1
}
```

Ustawienia

Ta klasa została zaprojektowana do pracy z typowymi ustawieniami administracyjnymi.

Odwołania do obiektu:

- Settings

Dostępne metody:

- settings.get - pobieranie ustawień
- settings.update - aktualizacja ustawień

Obiekt ustawień

Następujące obiekty są bezpośrednio powiązane z API `settings`.

Ustawienia

Obiekt ustawień ma następujące właściwości.

Property	Type	Description
default_lang	string	Domyślny język systemu. Domyślnie: `en_US`.
default_timezone	string	Domyślna strefa czasowa systemu. Domyślnie: `system` - domyślna strefa systemowa. Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP.
default_theme	string	Domyślny motyw. Możliwe wartości: `blue-theme` - *(domyślnie)* niebieski; `blue-classic-theme` - niebieski (klasyczny); `dark-theme` - ciemny; `dark-blue-theme` - ciemnoniebieski; `dark-classic-theme` - ciemny (klasyczny); `hc-light` - jasny o wysokim kontraście; `hc-dark` - ciemny o wysokim kontraście.
search_limit	integer	Limit wyników wyszukiwania i filtrowania. Domyślnie: 1000.
max_overview_table_size	integer	Maksymalna liczba kolumn i wierszy w widżetach pulpitu Przegląd danych i Przegląd wyzwalaczy. Domyślnie: 50.
max_in_table	integer	Maksymalna liczba elementów wyświetlanych w komórce tabeli. Domyślnie: 50.
server_check_interval	integer	Wyświetlaj ostrzeżenie, jeśli serwer Zabbix jest niedostępny. Możliwe wartości: 0 - Nie wyświetlaj ostrzeżenia; 10 - *(domyślnie)* Wyświetlaj ostrzeżenie.
work_period	string	Czas pracy. Domyślnie: 1-5,09:00-18:00.
show_technical_errors	integer	Wyświetlaj błędy techniczne (PHP/SQL) użytkownikom niebędącym administratorami Super oraz użytkownikom, którzy nie należą do grup użytkowników z włączonym trybem debugowania. Możliwe wartości: 0 - *(domyślnie)* Nie wyświetlaj błędów technicznych; 1 - Wyświetlaj błędy techniczne.
history_period	string	Maksymalny okres wyświetlania danych historycznych w widżetach pulpitu Najnowsze dane, Sieć i Przegląd danych. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`). Domyślnie: 24h.
period_default	string	Domyślny okres filtra czasu. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`, `1M`, `1y`), w tym miesiące i lata. Domyślnie: 1h.
max_period	string	Maksymalny okres dla filtra czasu. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`, `2h`, `1d`, `1M`, `1y`), w tym miesiące i lata. Domyślnie: 2y.

Property	Type	Description
severity_color_0	string	Kolor poziomu ważności „Niesklasyfikowane” w postaci szesnastkowego kodu koloru. Domyślnie: 97AAB3.
severity_color_1	string	Kolor poziomu ważności „Informacyjne” w postaci szesnastkowego kodu koloru. Domyślnie: 7499FF.
severity_color_2	string	Kolor poziomu ważności „Ostrzeżenie” w postaci szesnastkowego kodu koloru. Domyślnie: FFC859.
severity_color_3	string	Kolor poziomu ważności „Średnie” w postaci szesnastkowego kodu koloru. Domyślnie: FFA059.
severity_color_4	string	Kolor poziomu ważności „Wysokie” w postaci szesnastkowego kodu koloru. Domyślnie: E97659.
severity_color_5	string	Kolor poziomu ważności „Katastrofa” w postaci szesnastkowego kodu koloru. Domyślnie: E45959.
severity_name_0	string	Nazwa poziomu ważności „Niesklasyfikowane”. Domyślnie: Niesklasyfikowane.
severity_name_1	string	Nazwa poziomu ważności „Informacyjne”. Domyślnie: Informacyjne.
severity_name_2	string	Nazwa poziomu ważności „Ostrzeżenie”. Domyślnie: Ostrzeżenie.
severity_name_3	string	Nazwa poziomu ważności „Średnie”. Domyślnie: Średnie.
severity_name_4	string	Nazwa poziomu ważności „Wysokie”. Domyślnie: Wysokie.
severity_name_5	string	Nazwa poziomu ważności „Katastrofa”. Domyślnie: Katastrofa.
custom_color	integer	Używaj niestandardowych kolorów statusów zdarzeń. Możliwe wartości: 0 - *(domyślnie)* Nie używaj niestandardowych kolorów statusów zdarzeń; 1 - Używaj niestandardowych kolorów statusów zdarzeń.
ok_period	string	Wyświetlaj okres wyzwalaczy w stanie OK. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h). Domyślnie: 5m.
blink_period	string	Okres migania po zmianie stanu wyzwalacza. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h). Domyślnie: 2m.
problem_unack_color	string	Kolor niepotwierdzonych zdarzeń PROBLEM w postaci szesnastkowego kodu koloru. Domyślnie: CC0000.

Property	Type	Description
problem_ack_color	string	Kolor potwierdzonych zdarzeń PROBLEM w postaci szesnastkowego kodu koloru. Domyślnie: CC0000.
ok_unack_color	string	Kolor niepotwierdzonych zdarzeń RESOLVED w postaci szesnastkowego kodu koloru. Domyślnie: 009900.
ok_ack_color	string	Kolor potwierdzonych zdarzeń RESOLVED w postaci szesnastkowego kodu koloru. Domyślnie: 009900.
problem_unack_style	integer	Miganie niepotwierdzonych zdarzeń PROBLEM. Możliwe wartości: 0 - Nie pokazuj migania; 1 - *(domyślnie)* Pokazuj miganie.
problem_ack_style	integer	Miganie potwierdzonych zdarzeń PROBLEM. Możliwe wartości: 0 - Nie pokazuj migania; 1 - *(domyślnie)* Pokazuj miganie.
ok_unack_style	integer	Miganie niepotwierdzonych zdarzeń RESOLVED. Możliwe wartości: 0 - Nie pokazuj migania; 1 - *(domyślnie)* Pokazuj miganie.
ok_ack_style	integer	Miganie potwierdzonych zdarzeń RESOLVED. Możliwe wartości: 0 - Nie pokazuj migania; 1 - *(domyślnie)* Pokazuj miganie.
url	string	Adres URL frontendu.
discovery_groupid	ID	ID grupy hostów, do której automatycznie zostaną dodane wykryte hosty.
default_inventory_mode	integer	Domyślny tryb inwentaryzacji hosta. Możliwe wartości: -1 - *(domyślnie)* Wyłączony; 0 - Ręczny; 1 - Automatyczny.
alert_usrgrpid	ID	ID grupy użytkowników, do której zostanie wysłana wiadomość alarmowa o niedostępności bazy danych. Jeśli ustawiono wartość „0”, wiadomość alarmowa nie zostanie wysłana.
snmptrap_logging	integer	Rejestruj niedopasowane pułapki SNMP. Możliwe wartości: 0 - Nie rejestruj niedopasowanych pułapek SNMP; 1 - *(domyślnie)* Rejestruj niedopasowane pułapki SNMP.
login_attempts	integer	Liczba nieudanych prób logowania, po której formularz logowania zostanie zablokowany. Domyślnie: 5.
login_block	string	Okres, w którym formularz logowania będzie zablokowany, jeśli liczba nieudanych prób logowania przekroczy wartość zdefiniowaną w polu login_attempts. Akceptuje liczbę sekund (np. 1, 60) lub wartość z przyrostkiem czasu (np. 30s, 1m). Domyślnie: 30s.

Property	Type	Description
validate_uri_schemes	integer	Weryfikuj schematy URI. Możliwe wartości: 0 - Nie weryfikuj; 1 - *(domyślnie)* Weryfikuj.
uri_valid_schemes	string	Prawidłowe schematy URI. Domyślnie: http,https,ftp,file,mailto,tel,ssh.
x_frame_options	string	Nagłówek HTTP X-Frame-Options. Domyślnie: SAMEORIGIN.
iframe_sandboxing_enabled	integer	Używaj piaskownicy iframe. Możliwe wartości: 0 - Nie używaj; 1 - *(domyślnie)* Używaj.
iframe_sandboxing_exceptions	string	Wyjątki piaskownicy iframe.
connect_timeout	string	Limit czasu połączenia z serwerem Zabbix. Możliwy zakres wartości: 1-30s. Domyślnie: 3s. Zachowanie właściwości: - *required*
socket_timeout	string	Domyślny limit czasu sieciowego. Możliwy zakres wartości: 1-300s. Domyślnie: 3s. Zachowanie właściwości: - *required*
media_type_test_timeout	string	Limit czasu sieciowego testu typu nośnika. Możliwy zakres wartości: 1-300s. Domyślnie: 65s. Zachowanie właściwości: - *required*
item_test_timeout	string	Limit czasu sieciowego testów pozycji. Możliwy zakres wartości: 1-600s. Domyślnie: 60s. Zachowanie właściwości: - *required*
script_timeout	string	Limit czasu sieciowego wykonywania skryptu. Możliwy zakres wartości: 1-300s. Domyślnie: 60s. Zachowanie właściwości: - *required*

Property	Type	Description
report_test_timeout	string	Limit czasu sieciowego testu zaplanowanego raportu. Możliwy zakres wartości: 1-300s. Domyślnie: 60s. Zachowanie właściwości: - *required*
device_link_timeout	string	Limit czasu dodawania urządzenia mobilnego do użycia z aplikacją mobilną Zabbix; dotyczy komunikacji między API Zabbix a adapterem mostu. Możliwy zakres wartości: 1-300s. Domyślnie: 60s. Zachowanie właściwości: - *required*
auditlog_enabled	integer	Czy włączyć rejestrowanie audytu. Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
auditlog_mode	integer	Czy włączyć rejestrowanie audytu działań wykrywania niskiego poziomu, wykrywania sieci i automatycznej rejestracji wykonywanych przez serwer (użytkownik System). Możliwe wartości: 0 - Wyłącz; 1 - *(domyślnie)* Włącz.
ha_failover_delay	string	Opóźnienie przełączenia awaryjnego w sekundach. Domyślnie: 1m. Zachowanie właściwości: - *read-only*
geomaps_tile_provider	string	Dostawca kafelków mapy geograficznej. Możliwe wartości: `OpenStreetMap.Mapnik` - *(domyślnie)* OpenStreetMap Mapnik; `OpenTopoMap` - OpenTopoMap; `Stamen.TonerLite` - Stamen Toner Lite; `Stamen.Terrain` - Stamen Terrain; `USGS.USTopo` - USGS US Topo; `USGS.USImagery` - USGS US Imagery. Obsługuje pusty ciąg w celu określenia niestandardowych wartości `geomaps_tile_url`, `geomaps_max_zoom` i `geomaps_attribution`.
geomaps_tile_url	string	Adres URL kafelków mapy geograficznej. Zachowanie właściwości: - *supported* jeśli `geomaps_tile_provider` jest ustawione na pusty ciąg
geomaps_max_zoom	integer	Maksymalny poziom powiększenia mapy geograficznej. Możliwe wartości: 0-30. Zachowanie właściwości: - *supported* jeśli `geomaps_tile_provider` jest ustawione na pusty ciąg

Property	Type	Description
geomaps_attribution	string	Tekst atrybucji mapy geograficznej. Zachowanie właściwości: - *supported* jeśli `geomaps_tile_provider` jest ustawione na pusty ciąg
vault_provider	integer	Dostawca Vault. Możliwe wartości: 0 - *(domyślnie)* HashiCorp Vault; 1 - CyberArk Vault.
proxy_secrets_provider	integer	Rozwiązuj wartości makr sekretów przez: Możliwe wartości: 0 - *(domyślnie)* tylko serwer; 1 - serwer i proxy niezależnie.
timeout_zabbix_agent	string	Poświęcaj nie więcej niż `timeout_*` sekund na przetwarzanie. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. `30s`, `1m`). Akceptuje również makra użytkownika. Możliwy zakres wartości: 1-600s. Domyślnie: 3s. Domyślnie dla `timeout_browser`: 60s. Zachowanie właściwości: - *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		
timeout_telemetry_query		
apm_global_db	object	Konfiguracja globalnego źródła danych.

Globalne źródło danych

Obiekt globalnego źródła danych ma następujące właściwości:

Property	Type	Description
status	integer	Stan globalnego źródła danych. Możliwe wartości: 0 - *(domyślnie)* nieskonfigurowane; 1 - skonfigurowane.
url	string	Adres URL używany do łączenia ze źródłem danych. Zachowanie właściwości: - *wymagane*, jeśli użytkownik zmieni `status` na 1 (skonfigurowane).
authentication_type	integer	Typ uwierzytelniania. Możliwe wartości: 0 - *(domyślnie)* używana nazwa użytkownika i hasło; 1 - uwierzytelnianie nie jest używane. Zachowanie właściwości: - *wymagane*, jeśli użytkownik zmieni `status` na 1 (skonfigurowane).

Property	Type	Description
username	string	Nazwa użytkownika używana do logowania do źródła danych. Zachowanie właściwości: - *obsługiwane*, jeśli `status` ma wartość 1 (skonfigurowane), a `authentication_type` ma wartość 0 (nazwa użytkownika i hasło). - *wymagane*, jeśli użytkownik zmieni `authentication_type` na 0 (nazwa użytkownika i hasło).
password	string	Hasło używane do logowania do źródła danych. Wartość tego parametru może być pusta. Zachowanie właściwości: - *obsługiwane*, jeśli `status` ma wartość 1 (skonfigurowane), a `authentication_type` ma wartość 0 (nazwa użytkownika i hasło).
db	string	Nazwa bazy danych. Wartość tego parametru może być pusta. Zachowanie właściwości: - *obsługiwane*, jeśli `status` ma wartość 1 (skonfigurowane).
ssl_verify_peer	integer	Weryfikuj certyfikat serwera. Możliwe wartości: 0 - *(domyślnie)* wyłączone; 1 - włączone. Zachowanie właściwości: - *obsługiwane*, jeśli `status` ma wartość 1 (skonfigurowane), a `url` używa schematu `https://`.
ssl_verify_host	integer	Weryfikuj host. Możliwe wartości: 0 - wyłączone; 1 - *(domyślnie)* włączone. Zachowanie właściwości: - *obsługiwane*, jeśli `status` ma wartość 1 (skonfigurowane), a `url` używa schematu `https://`.

settings.get

Opis

`object settings.get(object parameters)`

Metoda umożliwia pobranie obiektu ustawień zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Typ	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Zwracane wartości

(object) Zwraca obiekt ustawień.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "settings.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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": "Niesklasyfikowane",
        "severity_name_1": "Informacje",
        "severity_name_2": "Ostrzeżenie",
        "severity_name_3": "Średnie",
        "severity_name_4": "Wysokie",
        "severity_name_5": "Katastrofa",
        "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
}
```


Źródło

CSettings::get() w *ui/include/classes/api/services/CSettings.php*.

settings.update

Opis

`object settings.update(object settings)`

Ta metoda umożliwia aktualizację istniejących wspólnych ustawień.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Właściwości ustawień do zaktualizowania.

Zwracane wartości

(`array`) Zwraca tablicę z nazwami zaktualizowanych parametrów.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "settings.update",
    "params": {
        "login_attempts": "1",
        "login_block": "1m"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        "login_attempts",
        "login_block"
    ],
    "id": 1
}
```

Źródło

CSettings::update() w *ui/include/classes/api/services/CSettings.php*.

Usługa

Ta klasa jest przeznaczona do pracy z usługami biznesowymi/infrastruktury IT.

Odwołania do obiektów:

- Usługa
- Reguła statusu
- Tag usługi
- Alarm usługi
- Tag problemu

Dostępne metody:

- service.create - utwórz nowe usługi
- service.delete - usuń usługi
- service.get - pobierz usługi
- service.update - zaktualizuj usługi

Obiekt usługi

Następujące obiekty są bezpośrednio powiązane z API `usługi`.

Usługa

Obiekt usługi ma następujące właściwości.

Property	Type	Description
serviceid	ID	ID usługi. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji

Property	Type	Description
algorithm	integer	Reguła obliczania stanu. Dotyczy tylko wtedy, gdy istnieją usługi podrzędne. Możliwe wartości: 0 - ustaw stan na OK; 1 - najbardziej krytyczny, jeśli wszystkie elementy podrzędne mają problemy; 2 - najbardziej krytyczny spośród usług podrzędnych. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa usługi. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
sortorder	integer	Pozycja usługi używana do sortowania. Możliwe wartości: 0-999. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
weight	integer	Waga usługi. Możliwe wartości: 0-1000000. Domyślnie: 0.
propagation_rule	integer	Reguła propagacji stanu. Możliwe wartości: 0 - *(domyślnie)* propaguj stan usługi bez zmian; 1 - zwiększ propagowany stan o podaną `propagation_value` (o 1 do 5 poziomów ważności); 2 - zmniejsz propagowany stan o podaną `propagation_value` (o 1 do 5 poziomów ważności); 3 - ignoruj tę usługę - stan nie jest w ogóle propagowany do usługi nadrzędnej; 4 - ustaw stały stan usługi, używając podanej `propagation_value`. Zachowanie właściwości: - *wymagane* jeśli ustawiono `propagation_value`

Property	Type	Description
propagation_value	integer	Wartość propagacji stanu. Możliwe wartości, jeśli `propagation_rule` jest ustawione na "0" lub "3": 0 - Nie sklasyfikowano. Możliwe wartości, jeśli `propagation_rule` jest ustawione na "1" lub "2": 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Możliwe wartości, jeśli `propagation_rule` jest ustawione na "4": -1 - OK; 0 - Nie sklasyfikowano; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Zachowanie właściwości: - *wymagane* jeśli ustawiono `propagation_rule`
status	integer	Czy usługa ma stan OK, czy stan problemu. Jeśli usługa ma stan problemu, `status` jest równy albo ważności najbardziej krytycznego problemu, albo najwyższemu statusowi usługi podrzędnej w stanie problemu. Jeśli usługa ma stan OK, `status` ma wartość -1. Zachowanie właściwości: - *tylko do odczytu*
description	string	Opis usługi.
uuid	string	Uniwersalny unikalny identyfikator, używany do powiązania importowanych usług z już istniejącymi. Generowany automatycznie, jeśli nie zostanie podany.
created_at	integer	Znacznik czasu Unix, kiedy usługa została utworzona.
readonly	integer	Dostęp do usługi. Możliwe wartości: 0 - Odczyt i zapis; 1 - Tylko do odczytu. Zachowanie właściwości: - *tylko do odczytu*

Reguła statusu

Obiekt reguły statusu ma następujące właściwości.

Właściwość	Typ	Opis
type	integer	Warunek ustawienia statusu (Nowy status). Możliwe wartości: 0 - jeśli co najmniej (N) usług podrzędnych ma status (Status) lub wyższy; 1 - jeśli co najmniej (N%) usług podrzędnych ma status (Status) lub wyższy; 2 - jeśli mniej niż (N) usług podrzędnych ma status (Status) lub niższy; 3 - jeśli mniej niż (N%) usług podrzędnych ma status (Status) lub niższy; 4 - jeśli waga usług podrzędnych o statusie (Status) lub wyższym wynosi co najmniej (W); 5 - jeśli waga usług podrzędnych o statusie (Status) lub wyższym wynosi co najmniej (N%); 6 - jeśli waga usług podrzędnych o statusie (Status) lub niższym jest mniejsza niż (W); 7 - jeśli waga usług podrzędnych o statusie (Status) lub niższym jest mniejsza niż (N%). Gdzie: - N (W) to `limit_value`; - (Status) to `limit_status`; - (Nowy status) to `new_status`. Zachowanie właściwości: - *wymagane*
limit_value	integer	Wartość graniczna. Możliwe wartości: - dla N i W: 1-100000; - dla N%: 1-100. Zachowanie właściwości: - *wymagane*
limit_status	integer	Status graniczny. Możliwe wartości: -1 - OK; 0 - Niesklasyfikowany; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Zachowanie właściwości: - *wymagane*
new_status	integer	Wartość nowego statusu. Możliwe wartości: 0 - Niesklasyfikowany; 1 - Informacja; 2 - Ostrzeżenie; 3 - Średni; 4 - Wysoki; 5 - Katastrofa. Zachowanie właściwości: - *wymagane*

Znacznik usługi

Obiekt znacznika usługi ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa znacznika usługi. Zachowanie właściwości: - *wymagane*
value	string	Wartość znacznika usługi.

Alarm usługi

Note:
Alarmów usługi nie można bezpośrednio tworzyć, aktualizować ani usuwać za pośrednictwem interfejsu API Zabbix.

Obiekty alarmu usługi reprezentują zmianę stanu usługi. Ma on następujące właściwości.

Property	Type	Description
clock	timestamp	Czas, w którym nastąpiła zmiana stanu usługi.
value	integer	Status usługi. Zapoznaj się z właściwością `status` usługi, aby uzyskać listę możliwych wartości.

Znacznik problemu

Znaczniki problemów umożliwiają łączenie usług ze zdarzeniami problemów. Obiekt znacznika problemu ma następujące właściwości.

Property	Type	Description
tag	string	Nazwa znacznika problemu. Zachowanie właściwości: - *wymagane*
operator	integer	Operator warunku mapowania. Możliwe wartości: 0 - *(domyślnie)* Równe; 2 - Zawiera.
value	string	Wartość znacznika problemu.

service.create

Opis

`object service.create(object/array services)`

Ta metoda umożliwia tworzenie nowych usług.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object/array)` usługi do utworzenia.

Oprócz standardowych właściwości usługi, metoda akceptuje następujące parametry.

Parametr	Typ	Opis
children	array	Podrzędne usługi, które mają zostać powiązane z usługą. Podrzędne usługi muszą mieć zdefiniowaną tylko właściwość `serviceid`.
parents	array	Nadrzędne usługi, które mają zostać powiązane z usługą. Nadrzędne usługi muszą mieć zdefiniowaną tylko właściwość `serviceid`.
tags	array	Tagi usługi, które mają zostać utworzone dla usługi.
problem_tags	array	Tagi problemów, które mają zostać utworzone dla usługi.
status_rules	array	Reguły statusu, które mają zostać utworzone dla usługi.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych usług we właściwości `serviceids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych usług.

Przykłady

Tworzenie usługi

Utwórz usługę, która zostanie przełączona w stan problemu, jeśli co najmniej jedno dziecko ma problem.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.create",
    "params": {
        "name": "Server 1",
        "algorithm": 1,
        "sortorder": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CService::create() w *ui/include/classes/api/services/CService.php*.

service.delete

Opis

`object service.delete(array serviceIds)`

Ta metoda umożliwia usuwanie usług.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(array) ID usług do usunięcia.

Zwracane wartości

(object) Zwraca obiekt zawierający identyfikatory usuniętych usług we właściwości serviceids.

Przykłady

Usuwanie wielu usług

Usuń dwie usługi.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.delete",
    "params": [
        "4",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CService::delete() w *ui/include/classes/api/services/CService.php*.

service.get

Opis

integer/array service.get(object parameters)

Metoda umożliwia pobieranie usług zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

(object) Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
serviceids	ID/array	Zwracaj tylko usługi o podanych identyfikatorach.
parentids	ID/array	Zwracaj tylko usługi, które są powiązane z podanymi usługami nadrzędnymi.
deep_parentids	flag	Zwracaj wszystkie bezpośrednie i pośrednie usługi podrzędne. Używane razem z parentids.
childids	ID/array	Zwracaj tylko usługi, które są powiązane z podanymi usługami podrzędnymi.

Parametr	Typ	Opis
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	object/array	Zwracaj tylko usługi z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie usługi. Możliwe wartości operator: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
problem_tags	object/array	Zwracaj tylko usługi z podanymi tagami problemów. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie usługi. Możliwe wartości operator: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
without_problem_tags	flag	Zwracaj tylko usługi bez tagów problemów.
slaids	ID/array	Zwracaj tylko usługi, które są powiązane z określonymi SLA.
selectChildren	query	Zwracaj właściwość `children` z usługami podrzędnymi. Obsługuje `count`.
selectParents	query	Zwracaj właściwość `parents` z usługami nadrzędnymi. Obsługuje `count`.
selectTags	query	Zwracaj właściwość `tags` z tagami usług. Obsługuje `count`.
selectProblemEvents	query	Zwracaj właściwość `problem_events` z tablicą obiektów zdarzeń problemów. Obiekt zdarzenia problemu ma następujące właściwości: `eventid` - `(ID)` ID zdarzenia; `severity` - `(string)` Bieżący poziom ważności zdarzenia; `name` - `(string)` Nazwa rozwiązanego zdarzenia. Obsługuje `count`.
selectProblemTags	query	Zwracaj właściwość `problem_tags` z tagami problemów. Obsługuje `count`.
selectStatusRules	query	Zwracaj właściwość `status_rules` z regułami statusu. Obsługuje `count`.

Parametr	Typ	Opis
selectStatusTimeline	object/array	Zwracaj właściwość `status_timeline` zawierającą zmiany stanu usługi dla podanych okresów. Format `[{"period_from": "<period_from>", "period_to": "<period_to>"}, ...]` - gdzie `period_from` jest datą początkową (włącznie; znacznik czasu integer), a `period_to` jest datą końcową (wyłącznie; znacznik czasu integer) dla interesującego Cię okresu. Zwraca tablicę wpisów zawierających właściwość `start_value` oraz tablicę `alarms` dla zmian stanu w określonych okresach.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `serviceid`, `name`, `status`, `sortorder`, `created_at`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wszystkich usług

Pobierz wszystkie dane o wszystkich usługach i ich zależnościach.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.get",
    "params": {
        "output": "extend",
        "selectChildren": "extend",
        "selectParents": ["serviceid", "name"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Źródło

CService::get() w *ui/include/classes/api/services/CService.php*.

service.update

Opis

`object service.update(object/array services)`

Ta metoda umożliwia aktualizację istniejących usług.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników , aby uzyskać więcej informacji.

Parametry

`(object/array)` właściwości usługi do zaktualizowania.

Właściwość `serviceid` musi być zdefiniowana dla każdej usługi, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości usługi, metoda akceptuje następujące parametry.

Parametr	Type	Opis
children	array	Podrzędne usługi, które zastąpią bieżące usługi podrzędne. Usługi podrzędne muszą mieć zdefiniowaną wyłącznie właściwość `serviceid`.
parents	array	Nadrzędne usługi, które zastąpią bieżące usługi nadrzędne. Usługi nadrzędne muszą mieć zdefiniowaną wyłącznie właściwość `serviceid`.
tags	array	Tagi usługi, które zastąpią bieżące tagi usługi.
problem_tags	array	Tagi problemów, które zastąpią bieżące tagi problemów.
status_rules	array	Reguły statusu, które zastąpią bieżące reguły statusu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych usług we właściwości `serviceids`.

Przykłady

Ustawianie usługi nadrzędnej dla usługi

Ustaw usługę o ID „3” jako usługę nadrzędną dla usługi o ID „5”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "5",
        "parents": [
            {
                "serviceid": "3"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "5"
        ]
    },
    "id": 1
}
```

Dodawanie zaplanowanego przestoju

Dodaj przestój dla usługi o ID „4”, zaplanowany co tydzień od poniedziałku 22:00 do wtorku 10:00.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "4",
        "times": [
            {
                "type": "1",
                "ts_from": "165600",
                "ts_to": "201600"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "serviceids": [
            "4"
        ]
    },
    "id": 1
}
```

Źródło

CService::update() w *ui/include/classes/api/services/CService.php*.

Uwierzytelnianie

Ta klasa została zaprojektowana do pracy z ustawieniami uwierzytelniania.

Odwołania do obiektów:

- Authentication

Dostępne metody:

- authentication.get - pobieranie ustawień uwierzytelniania
- authentication.update - aktualizowanie ustawień uwierzytelniania

Obiekt uwierzytelniania

Następujące obiekty są bezpośrednio powiązane z API `authentication`.

Uwierzytelnianie

Obiekt uwierzytelniania ma następujące właściwości.

Property	Type	Description
authentication_type	integer	Domyślne uwierzytelnianie. Możliwe wartości: 0 - *(domyślnie)* Wewnętrzne; 1 - LDAP.
http_auth_enabled	integer	Uwierzytelnianie HTTP. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone. Zachowanie właściwości: - *obsługiwane*, jeśli `$ZBX_FEATURE_FLAGS['http_auth_enabled']` ma wartość `true` w pliku konfiguracji frontend.
http_login_form	integer	Domyślny formularz logowania. Możliwe wartości: 0 - *(domyślnie)* Formularz logowania Zabbix; 1 - Formularz logowania HTTP. Zachowanie właściwości: - *obsługiwane*, jeśli `$ZBX_FEATURE_FLAGS['http_auth_enabled']` ma wartość `true` w pliku konfiguracji frontend.
http_strip_domains	string	Nazwa domeny do usunięcia. Zachowanie właściwości: - *obsługiwane*, jeśli `$ZBX_FEATURE_FLAGS['http_auth_enabled']` ma wartość `true` w pliku konfiguracji frontend.
http_case_sensitive	integer	Logowanie HTTP z uwzględnianiem wielkości liter. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone. Zachowanie właściwości: - *obsługiwane*, jeśli `$ZBX_FEATURE_FLAGS['http_auth_enabled']` ma wartość `true` w pliku konfiguracji frontend.
ldap_auth_enabled	integer	Uwierzytelnianie LDAP. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone.

Property	Type	Description
ldap_case_sensitive	integer	Logowanie LDAP z uwzględnianiem wielkości liter. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
ldap_userdirectoryid	ID	ID domyślnego katalogu użytkowników dla uwierzytelniania LDAP. Używany dla grup użytkowników z ustawieniem `gui_access` na LDAP lub Domyślne systemowe. Zachowanie właściwości: - *wymagane*, jeśli `ldap_auth_enabled` ma wartość „Włączone”
saml_auth_enabled	integer	Uwierzytelnianie SAML. Możliwe wartości: 0 - *(domyślnie)* Wyłączone; 1 - Włączone.
saml_case_sensitive	integer	Logowanie SAML z uwzględnianiem wielkości liter. Możliwe wartości: 0 - Wyłączone; 1 - *(domyślnie)* Włączone.
passwd_min_length	integer	Minimalna długość hasła. Zakres możliwych wartości wynosi od 1 do 70. Domyślnie: 8.
passwd_check_rules	integer	Reguły sprawdzania hasła. Możliwe wartości bitmapy: 0 - Sprawdzanie długości hasła; 1 - Sprawdzanie, czy hasło zawiera wielkie i małe litery alfabetu łacińskiego; 2 - Sprawdzanie, czy hasło zawiera cyfry; 4 - Sprawdzanie, czy hasło zawiera znaki specjalne; 8 - *(domyślnie)* Sprawdzanie, czy hasła nie ma na liście często używanych haseł oraz czy nie zawiera odmian słowa „Zabbix”, imienia, nazwiska ani nazwy użytkownika. To pole jest maską bitową; dopuszczalna jest dowolna suma możliwych wartości bitmapy (na przykład 15 oznacza sprawdzanie wszystkich reguł).
ldap_jit_status	integer	Stan aprowizacji LDAP. Możliwe wartości: 0 - Wyłączona dla skonfigurowanych IdP LDAP; 1 - Włączona dla skonfigurowanych IdP LDAP.
saml_jit_status	integer	Stan aprowizacji SAML. Możliwe wartości: 0 - Wyłączona dla skonfigurowanych IdP SAML; 1 - Włączona dla skonfigurowanych IdP SAML.
jit_provision_interval	string	Odstęp czasu między żądaniami aprowizacji JIT dla zalogowanego użytkownika. Akceptuje liczbę sekund (np. 3600) lub wartość z przyrostkiem czasu (np. `3600s`, `60m`, `1h`, `1d`, `1M`, `1y`), w tym miesiące i lata. Minimalna wartość: 1h. Domyślnie: 1h. Dostępne tylko dla aprowizacji LDAP.

Property	Type	Description
disabled_usrgrpid	ID	ID grupy użytkowników, do której należy przypisać użytkownika pozbawionego aprowizacji. Grupa użytkowników musi być wyłączona i po skonfigurowaniu nie można jej włączyć ani usunąć. Zachowanie właściwości: - *wymagane*, jeśli `ldap_jit_status` ma wartość „Włączona dla skonfigurowanych IdP LDAP” lub `saml_jit_status` ma wartość „Włączona dla skonfigurowanych IdP SAML”
mfa_status	integer	Uwierzytelnianie wieloskładnikowe. Możliwe wartości: 0 - Wyłączone (dla wszystkich skonfigurowanych metod MFA); 1 - Włączone (dla wszystkich skonfigurowanych metod MFA).
mfaid	ID	Domyślna metoda MFA dla grup użytkowników z włączonym MFA. Zachowanie właściwości: - *wymagane*, jeśli `mfa_status` ma wartość „Włączone”

authentication.get

Opis

`object authentication.get(object parameters)`

Metoda umożliwia pobranie obiektu uwierzytelniania zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje tylko jeden parametr.

Parametr	Type	Opis
output	query	Ten parametr jest opisany w komentarzu referencyjnym.

Zwracane wartości

`(object)` Zwraca obiekt uwierzytelniania.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Źródło

CAuthentication::get() w *ui/include/classes/api/services/CAuthentication.php*.

authentication.update

Opis

`object authentication.update(object authentication)`

Ta metoda umożliwia aktualizację istniejących ustawień uwierzytelniania.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Parametry

`(object)` Właściwości uwierzytelniania, które mają zostać zaktualizowane.

Zwracane wartości

`(array)` Zwraca tablicę z nazwami zaktualizowanych parametrów.

Przykłady

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "authentication.update",
    "params": {
        "http_auth_enabled": 1,
        "http_case_sensitive": 0,
        "http_login_form": 1
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        "http_auth_enabled",
        "http_case_sensitive",
```

```
        "http_login_form"
    ],
    "id": 1
}
```

Źródło

CAuthentication::update() w *ui/include/classes/api/services/CAuthentication.php*.

Użytkownik

Ta klasa została zaprojektowana do pracy z użytkownikami.

Odwołania do obiektów:

- Użytkownik
- Media

Dostępne metody:

- user.checkauthentication - sprawdzanie i przedłużanie sesji użytkownika
- user.create - tworzenie nowych użytkowników
- user.delete - usuwanie użytkowników
- user.get - pobieranie użytkowników
- user.login - logowanie do API
- user.logout - wylogowanie z API
- user.provision - provisionowanie użytkowników LDAP
- user.resettotp - resetowanie sekretów TOTP użytkownika
- user.unblock - odblokowywanie użytkowników
- user.update - aktualizowanie użytkowników

Obiekt użytkownika

Następujące obiekty są bezpośrednio powiązane z API `user`.

Użytkownik

Obiekt użytkownika ma następujące właściwości.

Property	Type	Description
userid	ID	ID użytkownika. Zachowanie właściwości: - *tylko do odczytu* - *wymagana* w operacjach aktualizacji
username	string	Nazwa użytkownika. Zachowanie właściwości: - *wymagana* w operacjach tworzenia - *tylko do odczytu* dla użytkowników aprowizowanych, jeśli użytkownik jest powiązany z katalogiem użytkowników (`userdirectoryid` ma ustawioną prawidłową wartość inną niż „0”), aprowizowanie w katalogu użytkowników jest włączone (`provision_status` obiektu User directory ma wartość „1”), a uwierzytelnianie dla całego aprowizowania LDAP lub SAML jest włączone (`ldap_jit_status` obiektu Authentication ma wartość „Enabled for configured LDAP IdPs” lub `saml_jit_status` obiektu Authentication ma wartość „Enabled for configured SAML IdPs”)
passwd	string	Hasło użytkownika. Wartość tego parametru może być pustym ciągiem, jeśli użytkownik jest powiązany z katalogiem użytkowników. Zachowanie właściwości: - *tylko do zapisu*

Property	Type	Description
roleid	ID	ID roli użytkownika. Należy pamiętać, że użytkownicy bez roli mogą logować się do Zabbix tylko przy użyciu uwierzytelniania LDAP lub SAML, pod warunkiem że ich informacje LDAP/SAML odpowiadają mapowaniom grup użytkowników skonfigurowanym w Zabbix.
attempt_clock	timestamp	Czas ostatniej nieudanej próby logowania. Zachowanie właściwości: - *tylko do odczytu*
attempt_failed	integer	Liczba ostatnich nieudanych prób logowania. Zachowanie właściwości: - *tylko do odczytu*
attempt_ip	string	Adres IP, z którego pochodziła ostatnia nieudana próba logowania. Zachowanie właściwości: - *tylko do odczytu*
autologin	integer	Określa, czy włączyć automatyczne logowanie. Możliwe wartości: 0 - *(domyślnie)* automatyczne logowanie wyłączone; 1 - automatyczne logowanie włączone.
autologout	string	Czas życia sesji użytkownika. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 2h). Jeśli ustawiono 0s, sesja nigdy nie wygaśnie. Domyślnie: 15m.
default_maintenance_period	string	Domyślny okres utrzymania dla utrzymania utworzonego za pomocą skrótu. Akceptuje sekundy lub jednostkę czasu z przyrostkiem (300s, 5m, 1h, 1d, 1w). Podana wartość zostanie zaokrąglona w dół do pełnych minut. Domyślnie: 1h.
lang	string	Kod języka użytkownika, na przykład `en_US`. Domyślnie: `default` - domyślny język systemu.
name	string	Imię użytkownika.
provisioned	integer	Określa, czy użytkownik został aprovisionowany. Możliwe wartości: 0 - nieaprovisionowany; 1 - aprovisionowany. Zachowanie właściwości: - *tylko do odczytu*
refresh	string	Automatyczny okres odświeżania. Akceptuje liczbę sekund (np. 1, 60, 300) lub wartość z przyrostkiem czasu (np. 30s, 1m, 1h). Domyślnie: 30s.
rows_per_page	integer	Liczba wierszy obiektów wyświetlanych na stronie. Domyślnie: 50.
surname	string	Nazwisko użytkownika.

Property	Type	Description
theme	string	Motyw użytkownika. Możliwe wartości: `default` - *(domyślnie)* domyślny motyw systemu; `blue-theme` - niebieski; `blue-classic-theme` - niebieski (klasyczny); `dark-theme` - ciemny; `dark-blue-theme` - ciemnoniebieski; `dark-classic-theme` - ciemny (klasyczny); `hc-light` - jasny, o wysokim kontraście; `hc-dark` - ciemny, o wysokim kontraście.
ts_provisioned	timestamp	Czas wykonania najnowszej operacji aprowizowania. Zachowanie właściwości: - *tylko do odczytu* - *obsługiwana* dla użytkowników typu *Super admin*
url	string	URL strony, na którą użytkownik zostanie przekierowany po zalogowaniu.
userdirectoryid	ID	ID katalogu użytkowników, z którym powiązany jest użytkownik. Używane do aprowizowania (tworzenia lub aktualizowania), a także do logowania użytkownika powiązanego z katalogiem użytkowników. W przypadku operacji logowania wartość tej właściwości ma pierwszeństwo przed właściwością `userdirectoryid` grup użytkowników, do których należy użytkownik. Domyślnie: 0. Zachowanie właściwości: - *tylko do odczytu* - *obsługiwana* dla użytkowników typu *Super admin*
timezone	string	Strefa czasowa użytkownika, na przykład `Europe/London`, `UTC`. Domyślnie: `default` - domyślna strefa czasowa systemu. Pełną listę obsługiwanych stref czasowych można znaleźć w dokumentacji PHP.

Media

Obiekt media ma następujące właściwości.

Property	Type	Description
mediaid	ID	ID media użytkownika. Zachowanie właściwości: - *tylko do odczytu*
mediatypeid	ID	ID typu media używanego przez media użytkownika. Zachowanie właściwości: - *wymagane*

Property	Type	Description
sendto	string/array	Adres, nazwa użytkownika lub inny identyfikator odbiorcy. Jeśli właściwość `type` obiektu Media type ma wartość „Email” lub „Push”, wartości są reprezentowane jako tablica. W przypadku „Push” tablica może zawierać symbol wieloznaczny (*) albo co najmniej jeden identyfikator urządzenia mobilnego. Jeśli właściwość `type` obiektu Media type ma wartość „Script”, „SMS” lub „Webhook”, wartość jest reprezentowana jako ciąg znaków. Zachowanie właściwości: - *wymagane*
active	integer	Określa, czy media są włączone. Możliwe wartości: 0 - *(domyślnie)* włączone; 1 - wyłączone.
severity	integer	Poziomy ważności wyzwalaczy, dla których mają być wysyłane powiadomienia. Możliwe wartości bitmapy: 1 - Niesklasyfikowane; 2 - Informacyjne; 4 - Ostrzeżenie; 8 - Średnie; 16 - Wysokie; 32 - Katastrofalne. To pole jest maską bitową; dopuszczalna jest dowolna suma możliwych wartości bitmapy (na przykład 48 dla poziomów Średnie, Wysokie i Katastrofalne). Wartość domyślna: 63.
period	string	Czas, w którym można wysyłać powiadomienia, określony jako okres czasu lub makra użytkownika rozdzielone średnikiem. Wartość domyślna: 1-7,00:00-24:00.
provisioned	integer	Określa, czy użytkownik został zainicjowany. Możliwe wartości: 0 - niezainicjowany; 1 - zainicjowany.
userdirectory_mediaid	ID	ID mapowania media w katalogu użytkowników dla zainicjowanych mediów. Zachowanie właściwości: - *tylko do odczytu* - *obsługiwane* w przypadku użytkowników typu *Super admin*

user.checkAuthentication

Opis

`object user.checkAuthentication`

Ta metoda sprawdza i przedłuża sesję użytkownika.

Attention:
Wywołanie metody `user.checkAuthentication` z użyciem parametru `sessionid` domyślnie przedłuża sesję użytkownika.

Parametry

Metoda przyjmuje następujące parametry.

Parameter	Type	Description
extend	boolean	Czy przedłużyć sesję użytkownika. Wartość domyślna: ”true”. Ustawienie wartości ”false” pozwala sprawdzić sesję użytkownika bez jej przedłużania.
sessionid	string	Zachowanie parametru: - *supported* jeśli `sessionid` jest ustawiony Token uwierzytelniania użytkownika authentication token. Zachowanie parametru: - *required* jeśli `token` nie jest ustawiony
token	string	Token API użytkownika API token. Zachowanie parametru: - *required* jeśli `sessionid` nie jest ustawiony

Zwracane wartości

`(object)` Zwraca obiekt zawierający informacje o użytkowniku.

Oprócz standardowych właściwości użytkownika zwracane są następujące informacje.

Właściwość	Type	Opis
auth_type	integer	Domyślne uwierzytelnianie dla użytkownika. Listę możliwych wartości można znaleźć we właściwości `authentication_type` obiektu Authentication object.
debug_mode	integer	Określa, czy tryb debugowania jest włączony czy wyłączony dla użytkownika. Listę możliwych wartości można znaleźć we właściwości `debug_mode` obiektu User group object.
deprovisioned	boolean	Określa, czy użytkownik należy do grupy wycofanych użytkowników.
gui_access	string	Metoda uwierzytelniania użytkownika do frontend. Listę możliwych wartości można znaleźć we właściwości `gui_access` obiektu User group object.
secret	string	Losowy ciąg 32 znaków. Jest generowany podczas logowania użytkownika. Właściwość `secret` nie jest zwracana, jeśli sesja użytkownika jest sprawdzana przy użyciu tokena API.
sessionid	string	Token uwierzytelniający, który musi być używany w kolejnych żądaniach API. Właściwość `sessionid` nie jest zwracana, jeśli sesja użytkownika jest sprawdzana przy użyciu tokena API.
type	integer	Typ użytkownika. Listę możliwych wartości można znaleźć we właściwości `type` obiektu Role object.
userip	string	Adres IP użytkownika.

Przykłady

Sprawdzenie uwierzytelnienia przy użyciu tokenu uwierzytelniającego

Sprawdź i przedłuż sesję użytkownika przy użyciu tokenu uwierzytelniającego użytkownika oraz zwróć dodatkowe informacje o użytkowniku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Sprawdzenie uwierzytelnienia przy użyciu tokena API

Sprawdź sesję użytkownika przy użyciu tokena API użytkownika i zwróć dodatkowe informacje o użytkowniku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.checkAuthentication",
    "params": {
        "token": "00aff470e07c12d707e50d98cfe39edef9e6ec349c14728dbdfbc8ddc5ea3eae"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Źródło

CUser::checkAuthentication() w *ui/include/classes/api/services/CUser.php*.

user.create

Opis

`object user.create(object/array users)`

Ta metoda umożliwia utworzenie nowych użytkowników.

Note:
Ta metoda jest dostępna tylko dla użytkowników typu *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach ról użytkowników. Więcej informacji zawiera sekcja Role użytkowników.

Note:
Siła hasła użytkownika jest sprawdzana zgodnie z regułami zasad haseł zdefiniowanymi przez Authentication API. Więcej informacji zawiera sekcja obiekt authentication.

Parametry

`(object/array)` Użytkownicy do utworzenia.

Oprócz standardowych właściwości użytkownika metoda akceptuje następujące parametry.

Parametr	Type	Opis
usrgrps	array	Grupy użytkowników, do których użytkownik ma zostać dodany. Grupy użytkowników muszą mieć zdefiniowaną tylko właściwość `usrgrpid`.

Parametr	Type	Opis
medias	array	Media użytkownika, które mają zostać utworzone.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych użytkowników we właściwości `userids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych użytkowników.

Przykłady

Tworzenie użytkownika

Utwórz nowego użytkownika, dodaj go do grupy użytkowników i utwórz dla niego nowe medium.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uwierzytelnianie
- Media
- Grupa użytkowników
- Rola

Źródło

CUser::create() w *ui/include/classes/api/services/CUser.php*.

user.delete

Opis

`object user.delete(array users)`

Ta metoda umożliwia usuwanie użytkowników.

> **Note:**
> Ta metoda jest dostępna tylko dla typu użytkownika *Super admin*. Uprawnienia do wywołania metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników .

Parametry

`(array)` ID użytkowników do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych użytkowników w we właściwości `userids`.

Przykłady

Usunięcie wielu użytkowników

Usunięcie dwóch użytkowników

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.delete",
    "params": [
        "1",
        "5"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CUser::delete() w *ui/include/classes/api/services/CUser.php*.

user.get

Opis

`integer/array user.get(object parameters)`

Ta metoda umożliwia pobieranie użytkowników zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

> **Note:**
> Podczas żądania informacji o mediach użytkownika, uprawnieniach lub roli, użytkownicy typu *Admin* i *User* mogą pobierać dane tylko o swoim własnym użytkowniku. Przykład znajduje się w sekcji Pobieranie użytkowników jako *Admin*.

Parametr	Typ	Opis
mediaids	ID/array	Zwracaj tylko użytkowników, którzy używają podanych mediów.
mediatypeids	ID/array	Zwracaj tylko użytkowników, którzy używają podanych typów mediów.
userids	ID/array	Zwracaj tylko użytkowników o podanych ID.
usrgrpids	ID/array	Zwracaj tylko użytkowników należących do podanych grup użytkowników.
getAccess	flag	Dodaje dodatkowe informacje o uprawnieniach użytkownika. Dodaje następujące właściwości dla każdego użytkownika: `gui_access` - `(integer)` metoda uwierzytelniania użytkownika we frontendzie. Listę możliwych wartości można znaleźć we właściwości `gui_access` obiektu grupy użytkowników. `debug_mode` - `(integer)` wskazuje, czy debugowanie jest włączone dla użytkownika. Możliwe wartości: 0 - debugowanie wyłączone, 1 - debugowanie włączone. `users_status` - `(integer)` wskazuje, czy użytkownik jest wyłączony. Możliwe wartości: 0 - użytkownik włączony, 1 - użytkownik wyłączony.
selectMedias	query	Zwracaj media używane przez użytkownika we właściwości `medias`.
selectMediatypes	query	Zwracaj typy mediów używane przez użytkownika we właściwości `mediatypes`. Zobacz `mediatype.get`, aby poznać ograniczenia zależne od typu użytkownika.
selectUsrgrps	query	Zwracaj grupy użytkowników, do których należy użytkownik, we właściwości `usrgrps`. Zobacz `usergroup.get`, aby poznać ograniczenia zależne od typu użytkownika.
selectRole	query	Zwracaj rolę użytkownika we właściwości `role`.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Możliwe właściwości obiektu User dla użytkowników typu *Admin* i *User* podczas żądania danych o użytkownikach z ich grupy użytkowników: `userid`, `name`, `surname`, `username`.
output	query	Właściwości obiektu User, które mają zostać zwrócone. Użytkownicy typu *Admin* i *User* mogą pobierać tylko następujące właściwości: - Dla własnego użytkownika: `userid`, `attempt_clock`, `attempt_failed`, `attempt_ip`, `autologin`, `autologout`, `lang`, `name`, `provisioned`, `refresh`, `roleid`, `rows_per_page`, `surname`, `theme`, `timezone`, `url`, `username`. - Dla użytkowników z ich grupy użytkowników: `userid`, `name`, `surname`, `username`. Domyślnie: `extend`.

Parametr	Typ	Opis
search	object	Zwracaj wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%…%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Możliwe właściwości obiektu User dla użytkowników typu *Admin* i *User* podczas żądania danych o użytkownikach z ich grupy użytkowników: `name`, `surname`, `username`.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `userid`, `username`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
preservekeys	boolean	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie użytkowników

Pobierz wszystkich skonfigurowanych użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Pobieranie użytkowników jako *Admin*

Jako użytkownik typu *Admin* pobierz szczegółowe dane dotyczące własnego użytkownika oraz ograniczone dane użytkowników należących do Twojej grupy użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": "extend",
        "getAccess": true,
        "selectMedias": "extend",
        "selectMediatypes": "extend",
        "selectUsrgrps": "extend",
        "selectRole": "extend"
    },
    "id": 1
}
```

Odpowiedź:


```
{
  "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",
        "ts_provisioned": "0",
        "provisioned": "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
}
```

Pobieranie danych użytkownika

Pobierz dane użytkownika o identyfikatorze „2”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.get",
    "params": {
        "output": ["userid", "username"],
        "selectRole": "extend",
        "userids": "2"
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "userid": "2",
      "username": "guest",
      "role": {
        "roleid": "4",
        "name": "Guest role",
        "type": "1",
        "readonly": "0"
      }
    }
  ],
  "id": 1
}
```

Zobacz także

- Media
- Typ nośnika
- Grupa użytkowników
- Rola

Źródło

CUser::get() w *ui/include/classes/api/services/CUser.php*.

user.login

Opis

`string/object user.login(object parameters)`

Ta metoda umożliwia zalogowanie się do API i wygenerowanie tokenu uwierzytelniającego.

Warning:
Podczas korzystania z tej metody należy również wykonać user.logout, aby zapobiec tworzeniu dużej liczby otwartych rekordów sesji.

Attention:
Ta metoda jest dostępna tylko dla nieuwierzytelnionych użytkowników, którzy nie należą do żadnej grupy użytkowników z włączonym uwierzytelnianiem wieloskładnikowym.

Parametry

`(object)` Parametry zawierające nazwę użytkownika i hasło.

Metoda akceptuje następujące parametry.

Parametr	Typ	Opis
password	string	Hasło użytkownika. Zachowanie parametru: - *wymagany*
username	string	Nazwa użytkownika. Zachowanie parametru: - *wymagany*
userData	flag	Zwraca informacje o uwierzytelnionym użytkowniku.

Zwracane wartości

`(string/object)` Jeśli używany jest parametr `userData`, zwracany jest obiekt zawierający informacje o uwierzytelnionym użytkowniku.

Oprócz standardowych właściwości użytkownika zwracane są następujące informacje:

Właściwość	Typ	Opis
auth_type	integer	Domyślna metoda uwierzytelniania dla użytkownika. Listę możliwych wartości można znaleźć we właściwości `authentication_type` obiektu Authentication object.
debug_mode	integer	Określa, czy tryb debugowania jest włączony czy wyłączony dla użytkownika. Listę możliwych wartości można znaleźć we właściwości `debug_mode` obiektu User group object.
deprovisioned	boolean	Określa, czy użytkownik należy do grupy wycofanych użytkowników.
gui_access	string	Metoda uwierzytelniania użytkownika do frontendu. Listę możliwych wartości można znaleźć we właściwości `gui_access` obiektu User group object.
mfaid	integer	ID metody MFA, która ma być używana dla użytkownika podczas logowania. Zwraca "0", jeśli MFA jest globalnie wyłączone lub wyłączone dla wszystkich grup użytkowników, do których należy użytkownik.
secret	string	Losowy ciąg 32 znaków. Jest generowany podczas logowania użytkownika.
sessionid	string	Token uwierzytelniający, który musi być używany w kolejnych żądaniach API.
type	integer	Typ użytkownika. Listę możliwych wartości można znaleźć we właściwości `type` obiektu Role object.
userip	string	Adres IP użytkownika.

Note:
Jeśli użytkownik został pomyślnie uwierzytelniony po jednej lub większej liczbie nieudanych prób, metoda zwróci bieżące wartości właściwości `attempt_clock`, `attempt_failed` i `attempt_ip`, a następnie je zresetuje.

Jeśli parametr `userData` nie jest używany, metoda zwraca token uwierzytelniający wymagany do uwierzytelniania.

Przykłady

Uwierzytelnianie użytkownika

Uwierzytelnij użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "0424bd59b807674191e7d77572075f33",
    "id": 1
}
```

Żądanie informacji o uwierzytelnionym użytkowniku

Uwierzytelnij i zwróć dodatkowe informacje o użytkowniku.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.login",
    "params": {
        "username": "Admin",
        "password": "zabbix",
        "userData": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```


Zobacz także

- user.logout

Źródło

CUser::login() w *ui/include/classes/api/services/CUser.php*.

user.logout

Opis

`string/object user.logout(array)`

Ta metoda umożliwia wylogowanie się z API i unieważnia bieżący token uwierzytelniania.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` Metoda akceptuje pustą tablicę.

Zwracane wartości

`(boolean)` Zwraca `true`, jeśli użytkownik został pomyślnie wylogowany.

Przykłady

Wylogowanie

Wyloguj się z API.

Żądanie:


```
{
    "jsonrpc": "2.0",
    "method": "user.logout",
    "params": [],
    "id": 1
}
```


Odpowiedź:


```
{
    "jsonrpc": "2.0",
    "result": true,
    "id": 1
}
```


Zobacz także

- user.login

Źródło

CUser::login() w *ui/include/classes/api/services/CUser.php*.

user.provision

Opis

`object user.provision(object/array users)`

Ta metoda umożliwia aprowizację użytkowników LDAP.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID użytkowników do provisioningu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory provisioned users we właściwości `userids`.

Przykłady

Provisionowanie wielu użytkowników

Provisionuj dwóch użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.provision",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CUser::provision() w *ui/include/classes/api/services/CUser.php*.

user.resettotp

Opis

`object user.resettotp(object/array users)`

Ta metoda umożliwia zresetowanie sekretów TOTP użytkownika.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID użytkowników, dla których należy zresetować sekrety TOTP.

Note:
Sesje użytkowników dla wskazanych użytkowników również zostaną usunięte (z wyjątkiem użytkownika wysyłającego żądanie).

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory użytkowników, dla których sekrety TOTP zostały zresetowane, we właściwości `userids`.

Przykłady

Resetowanie sekretów TOTP dla wielu użytkowników

Zresetuj sekrety TOTP dla dwóch użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.resettotp",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Zobacz także

- Obiekt MFA

Źródło

CUser::resettotp() w *ui/include/classes/api/services/CUser.php*.

user.unblock

Opis

`object user.unblock(array userids)`

Ta metoda umożliwia odblokowanie użytkowników.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID użytkowników do odblokowania.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory odblokowanych użytkowników we właściwości `userids`.

Przykłady

Odblokowywanie wielu użytkowników

Odblokuj dwóch użytkowników.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.unblock",
    "params": [
        "1",
        "5"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1",
            "5"
        ]
    },
    "id": 1
}
```

Źródło

CUser::unblock() w *ui/include/classes/api/services/CUser.php*.

user.update

Opis

`object user.update(object/array users)`

Ta metoda umożliwia aktualizację istniejących użytkowników.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody mogą zostać odebrane w ustawieniach roli użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Note:
Siła hasła użytkownika jest sprawdzana zgodnie z regułami polityki haseł zdefiniowanymi przez Authentication API. Zobacz Authentication API aby uzyskać więcej informacji.

Parametry

`(object/array)` Właściwości użytkownika do zaktualizowania.

Właściwość `userid` musi być zdefiniowana dla każdego użytkownika, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości użytkownika metoda akceptuje następujące parametry.

Parametr	Typ	Opis
current_passwd	string	Bieżące hasło użytkownika. Wartość tego parametru może być pustym ciągiem znaków, jeśli użytkownik jest powiązany z katalogiem użytkowników. Zachowanie parametru: - *tylko do zapisu* - *wymagany*, jeśli `passwd` obiektu User object jest ustawione i użytkownik zmienia własne hasło
usrgrps	array	Grupy użytkowników zastępujące istniejące grupy użytkownika. Dla grup użytkowników musi być zdefiniowana tylko właściwość `usrgrpid`.
medias	array	Media użytkownika zastępujące istniejące, nieudostępnione media. Udostępnione media można pominąć podczas aktualizacji mediów.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych użytkowników we właściwości `userids`.

Przykłady

Zmiana nazwy użytkownika

Zmień nazwę użytkownika na John Doe.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "1",
        "name": "John",
        "surname": "Doe"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "1"
        ]
    },
    "id": 1
}
```

Zmiana roli użytkownika

Zmień rolę użytkownika.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "user.update",
    "params": {
        "userid": "12",
        "roleid": "6"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "userids": [
            "12"
        ]
    },
    "id": 1
}
```

Zobacz także

- Uwierzytelnianie

Źródło

CUser::update() w *ui/include/classes/api/services/CUser.php*.

Wykres

Ta klasa jest przeznaczona do pracy z wykresami.

Odniesienia do obiektów:

- Wykres

Dostępne metody:

- graph.create - tworzenie nowego wykresu
- graph.delete - usuwanie wykresu
- graph.get - pobieranie wykresu
- graph.update - aktualizacja wykresu

Obiekt Graph

Następujące obiekty są bezpośrednio powiązane z API `graph`.

Wykres

Obiekt wykresu ma następujące właściwości.

Właściwość	Typ	Opis
graphid	ID	ID wykresu. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
height	integer	Wysokość wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
name	string	Nazwa wykresu. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
width	integer	Szerokość wykresu w pikselach. Zachowanie właściwości: - *wymagane* dla operacji tworzenia

Właściwość	Typ	Opis
flags	integer	Pochodzenie wykresu. Możliwe wartości: 0 - *(domyślnie)* zwykły wykres; 4 - wykres przekonwertowany z prototypu. Zachowanie właściwości: - *tylko do odczytu*
graphtype	integer	Typ układu wykresu. Możliwe wartości: 0 - *(domyślnie)* normalny; 1 - skumulowany; 2 - kołowy; 3 - rozstrzelony.
percent_left	float	Lewy percentyl. Domyślnie: 0.
percent_right	float	Prawy percentyl. Domyślnie: 0.
show_3d	integer	Czy wykresy kołowe i rozstrzelone mają być wyświetlane w 3D. Możliwe wartości: 0 - *(domyślnie)* wyświetlaj w 2D; 1 - wyświetlaj w 3D.
show_legend	integer	Czy wyświetlać legendę na wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
show_work_period	integer	Czy wyświetlać czas pracy na wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
show_triggers	integer	Czy wyświetlać linię wyzwalacza na wykresie. Możliwe wartości: 0 - ukryj; 1 - *(domyślnie)* wyświetlaj.
templateid	ID	ID wykresu nadrzędnego szablonu. Zachowanie właściwości: - *tylko do odczytu*
yaxismax	float	Stała maksymalna wartość osi Y. Domyślnie: 100.
yaxismin	float	Stała minimalna wartość osi Y. Domyślnie: 0.
ymax_itemid	ID	ID pozycji używanej jako maksymalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymax_type` było ustawione na „calculated”.
ymax_type	integer	Metoda obliczania wartości maksymalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.

Właściwość	Typ	Opis
ymin_itemid	ID	ID pozycji używanej jako minimalna wartość osi Y. Jeśli użytkownik nie ma dostępu do określonej pozycji, wykres jest renderowany tak, jakby `ymin_type` było ustawione na „calculated”.
ymin_type	integer	Metoda obliczania wartości minimalnej dla osi Y. Możliwe wartości: 0 - *(domyślnie)* calculated; 1 - stała; 2 - pozycja.
uuid	string	Uniwersalny unikalny identyfikator używany do łączenia importowanych wykresów z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane*, jeśli wykres należy do szablonu

graph.create

Opis

`object graph.create(object/array graphs)`

Ta metoda umożliwia tworzenie nowych wykresów.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Wykresy do utworzenia.

Oprócz standardowych właściwości wykresu, metoda akceptuje następujące parametry.

Parametr	Type	Opis
gitems	array	Pozycje wykresu, które mają zostać utworzone dla wykresu. Zachowanie parametru: - *wymagane*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych wykresów we właściwości `graphids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych wykresów.

Przykłady

Tworzenie wykresu

Utwórz wykres z dwiema pozycjami.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652"
        ]
    },
    "id": 1
}
```

Zobacz także

- Pozycja wykresu

Źródło

CGraph::create() w *ui/include/classes/api/services/CGraph.php*.

graph.delete

Opis

`object graph.delete(array graphIds)`

Ta metoda umożliwia usuwanie wykresów.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody mogą zostać cofnięte w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID wykresów do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych wykresów we właściwości `graphids`.

Przykłady

Usuwanie wielu wykresów

Usunięcie dwóch wykresów.

Request:

```
{
    "jsonrpc": "2.0",
    "method": "graph.delete",
    "params": [
        "652",
        "653"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
```

```
    "id": 1
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Źródło

CGraph::delete() w *ui/include/classes/api/services/CGraph.php*.

graph.get

Opis

`integer/array graph.get(object parameters)`

Metoda umożliwia pobieranie wykresów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników, aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
graphids	ID/array	Zwracaj tylko wykresy o podanych ID.
groupids	ID/array	Zwracaj tylko wykresy należące do hostów lub szablonów w podanych grupach hostów lub grupach szablonów.
templateids	ID/array	Zwracaj tylko wykresy należące do podanych szablonów.
hostids	ID/array	Zwracaj tylko wykresy należące do podanych hostów.
itemids	ID/array	Zwracaj tylko wykresy zawierające podane pozycje.
templated	boolean	Jeśli ustawiono `true`, zwracaj tylko wykresy należące do szablonów.
inherited	boolean	Jeśli ustawiono `true`, zwracaj tylko wykresy odziedziczone z szablonu.
expandName	flag	Rozwiń makra w nazwie wykresu.
selectDiscoveryData	query	Zwróć właściwość `discoveryData` z danymi obiektu wykrywania wykresu. Obiekt wykrywania wykresu łączy wykryty wykres z prototypem wykresu, z którego został wykryty. Ma następujące właściwości: `parent_graphid` - `(ID)` ID prototypu wykresu, z którego wykres został utworzony; `status` - `(int)` status wykrywania wykresu: 0 - *(domyślnie)* wykres jest wykryty, 1 - wykres nie jest już wykryty; `ts_delete` - `(timestamp)` czas, kiedy wykres, który nie jest już wykrywany, zostanie usunięty.
selectDiscoveryRule	query	Zwróć właściwość `discoveryRule` z regułą wykrywania niskiego poziomu, która utworzyła wykres.
selectHostGroups	query	Zwróć właściwość `hostgroups` z grupami hostów, do których należy wykres.

Parametr	Typ	Opis
selectTemplateGroups	query	Zwróć właściwość `templategroups` z grupami szablonów, do których należy wykres.
selectTemplates	query	Zwróć właściwość `templates` z szablonami, do których należy wykres.
selectHosts	query	Zwróć właściwość `hosts` z hostami, do których należy wykres.
selectItems	query	Zwróć właściwość `items` z pozycjami użytymi na wykresie.
selectGraphItems	query	Zwróć właściwość `gitems` z pozycjami użytymi na wykresie.
filter	object	Zwracaj tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym klucze są nazwami właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - techniczna nazwa hosta, do którego należy wykres; `hostid` - ID hosta, do którego należy wykres.
sortfield	string/array	Sortuj wynik według podanych właściwości. Możliwe wartości: `graphid`, `name`, `graphtype`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectGraphDiscovery	query	Zwróć właściwość `graphDiscovery` z obiektem wykrywania wykresu. Obiekty wykrywania wykresu łączą wykres z prototypem wykresu, z którego został utworzony. To zapytanie jest **przestarzałe**; użyj zamiast tego `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie wykresów z hostów

Pobierz wszystkie wykresy z hosta „10107” i posortuj je według nazwy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graph.get",
    "params": {
        "output": "extend",
        "hostids": 10107,
        "sortfield": "name"
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "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": "CPU load",
            "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": "CPU utilization",
            "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": "Disk space usage /",
            "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
}
```


Zobacz także

- Reguła wykrywania
- Pozycja wykresu
- Pozycja
- Host
- Grupa hostów
- Szablon
- Grupa szablonów

Źródło

CGraph::get() w *ui/include/classes/api/services/CGraph.php*.

graph.update

Opis

`object graph.update(object/array graphs)`

Ta metoda umożliwia aktualizację istniejących wykresów.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości wykresu do zaktualizowania.

Właściwość `graphid` musi być zdefiniowana dla każdego wykresu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości wykresu metoda akceptuje następujące parametry.

Parametr	Typ	Opis
gitems	array	Pozycje wykresu służące do zastąpienia istniejących pozycji wykresu. Jeśli pozycja wykresu ma zdefiniowaną właściwość `gitemid`, zostanie zaktualizowana; w przeciwnym razie zostanie utworzona nowa pozycja wykresu.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych wykresów we właściwości `graphids`.

Przykłady

Ustawianie maksimum skali Y

Ustaw maksimum skali Y na stałą wartość 100.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "graph.update",
    "params": {
        "graphid": "439",
        "ymax_type": 1,
        "yaxismax": 100
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "graphids": [
            "439"
        ]
    },
    "id": 1
}
```

Źródło

CGraph::update() w *ui/include/classes/api/services/CGraph.php*.

Wykryta usługa

Ta klasa została zaprojektowana do pracy z wykrytymi usługami.

Odwołania do obiektów:

- Wykryta usługa

Dostępne metody:

- dservice.get - pobieranie wykrytych usług

Obiekt odkrywania usługi

Następujące obiekty są bezpośrednio powiązane z API `dservice`.

Wykryta usługa

> **Note:**
> Wykryte usługi są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pomocą API.

Obiekt wykrytej usługi zawiera informacje o usłudze wykrytej przez regułę wykrywania sieci na hoście. Ma następujące właściwości.

Właściwość	Typ	Opis
dserviceid	ID	ID wykrytej usługi.
dcheckid	ID	ID kontroli wykrywania użytej do wykrycia usługi.
dhostid	ID	ID wykrytego hosta, na którym działa usługa.
dns	string	DNS hosta, na którym działa usługa.
ip	string	Adres IP hosta, na którym działa usługa.
lastdown	timestamp	Czas, kiedy wykryta usługa ostatnio przestała działać.
lastup	timestamp	Czas, kiedy wykryta usługa ostatnio zaczęła działać.
port	integer	Numer portu usługi.
status	integer	Status usługi. Możliwe wartości: 0 - usługa działa; 1 - usługa nie działa.
value	string	Wartość zwrócona przez usługę podczas wykonywania kontroli wykrywania agent, SNMPv1, SNMPv2 lub SNMPv3.

dservice.get

Opis

`integer/array dservice.get(object parameters)`

Metoda umożliwia pobranie wykrytych usług zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dserviceids	ID/array	Zwraca tylko wykryte usługi o podanych identyfikatorach.
dhostids	ID/array	Zwraca tylko wykryte usługi należące do podanych wykrytych hostów.
dcheckids	ID/array	Zwraca tylko wykryte usługi, które zostały wykryte przez podane kontrole wykrywania.
druleids	ID/array	Zwraca tylko wykryte usługi, które zostały wykryte przez podane reguły wykrywania.
selectDRules	query	Zwraca właściwość `drules` zawierającą tablicę reguł wykrywania, które wykryły usługę.
selectDHosts	query	Zwraca właściwość `dhosts` zawierającą tablicę wykrytych hostów, do których należy usługa.
selectHosts	query	Zwraca właściwość `hosts` zawierającą hosty z tym samym adresem IP i proxy co usługa. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wynik zostanie posortowany według `hostid`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dserviceid`, `dhostid`, `ip`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	

Parametr	Typ	Opis
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie usług wykrytych na hoście

Pobierz wszystkie wykryte usługi znalezione na wykrytym hoście „11”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "dservice.get",
    "params": {
        "output": "extend",
        "dhostids": "11"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Wykryty host
- Kontrola wykrywania
- Host

Źródło

CDService::get() w *ui/include/classes/api/services/CDService.php*.

Wykryty host

Ta klasa została zaprojektowana do pracy z wykrytymi hostami.

Odwołania do obiektów:

- Wykryty host

Dostępne metody:

- dhost.get - pobieranie wykrytych hostów

Obiekt odkrywania hosta

Następujące obiekty są bezpośrednio powiązane z API `dhost`.

Wykryty host

> **Note:**
> Wykryte hosty są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pośrednictwem API.

Obiekt wykrytego hosta zawiera informacje o hoście wykrytym przez regułę wykrywania sieci. Ma następujące właściwości.

Właściwość	Typ	Opis
dhostid	ID	ID wykrytego hosta.
druleid	ID	ID reguły wykrywania, która wykryła host.
lastdown	timestamp	Czas, kiedy wykryty host ostatnio przestał być dostępny.
lastup	timestamp	Czas, kiedy wykryty host ostatnio stał się dostępny.
status	integer	Czy wykryty host jest dostępny, czy niedostępny. Host jest dostępny, jeśli ma co najmniej jedną aktywną wykrytą usługę. Możliwe wartości: 0 - host dostępny; 1 - host niedostępny.

dhost.get

Opis

`integer/array dhost.get(object parameters)`

Ta metoda umożliwia pobieranie wykrytych hostów zgodnie z podanymi parametrami.

> **Note:**
> Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Zobacz Role użytkowników aby uzyskać więcej informacji.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
dhostids	ID/array	Zwraca tylko wykryte hosty o podanych identyfikatorach.
druleids	ID/array	Zwraca tylko wykryte hosty, które zostały utworzone przez podane reguły wykrywania.
dserviceids	ID/array	Zwraca tylko wykryte hosty, na których działają podane usługi.
selectDRules	query	Zwraca właściwość `drules` zawierającą tablicę reguł wykrywania, które wykryły host.
selectDServices	query	Zwraca właściwość `dservices` zawierającą wykryte usługi działające na hoście. Obsługuje `count`.
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Ma zastosowanie do następujących podzapytań: `selectDServices` - wyniki zostaną posortowane według `dserviceid`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `dhostid`, `druleid`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca jedną z opcji:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli parametr `countOutput` został wykorzystany.

Przykłady

Pobierz odkryte hosty według reguły wykrywania

Pobierz wszystkie hosty i odkryte usługi, które uruchamiają, które zostały wykryte przez regułę wykrywania „4”.

Zapytanie:

```
{
  "jsonrpc": "2.0",
  "method": "dhost.get",
  "params": {
    "output": "extend",
    "selectDServices": "extend",
    "druleids": "4"
  },
  "id": 1
}
```

Odpowiedź:

```
{
  "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
}
```


Zobacz także

- Wykryta usługa
- Reguła wykrywania

Źródło

CDHost::get() w *ui/include/classes/api/services/CDHost.php*.

Wyrażenie regularne

Ta klasa jest przeznaczona do pracy z globalnymi wyrażeniami regularnymi.

Odwołania do obiektów:

- Wyrażenie regularne
- Wyrażenia

Dostępne metody:

- regexp.create - tworzenie nowych wyrażeń regularnych
- regexp.delete - usuwanie wyrażeń regularnych
- regexp.get - pobieranie wyrażeń regularnych
- regexp.update - aktualizowanie wyrażeń regularnych

Obiekt wyrażenia regularnego

Następujące obiekty są bezpośrednio powiązane z API `regexp`.

Wyrażenie regularne

Globalny obiekt wyrażenia regularnego ma następujące właściwości.

Property	Type	Description
regexpid	ID	ID wyrażenia regularnego. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji.
name	string	Nazwa wyrażenia regularnego. Zachowanie właściwości: - *wymagane* dla operacji tworzenia.
description	text	Opis wyrażenia regularnego.
test_string	string	Ciąg testowy.

Wyrażenia

Obiekt expressions ma następujące właściwości.

Property	Type	Description
expression	string	Wyrażenie regularne. Property behavior: - *required*
expression_type	integer	Typ wyrażenia regularnego. Possible values: 0 - Zawiera ciąg znaków; 1 - Zawiera dowolny podciąg z listy; 2 - Nie zawiera ciągu znaków; 3 - Pasuje do wyrażenia regularnego; 4 - Nie pasuje do wyrażenia regularnego. Property behavior: - *required*
exp_delimiter	string	Separator wyrażenia. Default value: `","`. Possible values: `","` or `"."`, or `"/"`. Property behavior: - *supported* if `expression_type` is set to *Contains any substring from list*.
case_sensitive	integer	Rozróżnianie wielkości liter. Default value: 0. Possible values: 0 - Bez rozróżniania wielkości liter; 1 - Z rozróżnianiem wielkości liter.

regexp.create

Opis

`object regexp.create(object/array regularExpressions)`

Ta metoda umożliwia tworzenie nowych globalnych wyrażeń regularnych.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji znajdziesz w sekcji Role użytkowników.

Parametry

`(object/array)` Wyrażenia regularne do utworzenia.

Oprócz standardowych właściwości metoda akceptuje następujące parametry.

Parametr	Typ	Opis
expressions	array	Opcje wyrażeń. Zachowanie parametru: - *wymagany*

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych wyrażeń regularnych we właściwości `regexpids`.

Przykłady

Tworzenie nowego globalnego wyrażenia regularnego.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "9"
        ]
    },
    "id": 1
}
```

Źródło

CRegexp::create() w *ui/include/classes/api/services/CRegexp.php*.

regexp.delete

Opis

`object regexp.delete(array regexpids)`

Ta metoda umożliwia usuwanie globalnych wyrażeń regularnych.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID wyrażeń regularnych do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych wyrażeń regularnych we właściwości `regexpids`.

Przykłady

Usuwanie wielu globalnych wyrażeń regularnych.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.delete",
    "params": [
        "16",
        "17"
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "16",
            "17"
        ]
    },
    "id": 1
}
```

Źródło

CRegexp::delete() w *ui/include/classes/api/services/CRegexp.php*.

regexp.get

Opis

`integer/array regexp.get(object parameters)`

Metoda umożliwia pobieranie globalnych wyrażeń regularnych zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna wyłącznie dla *Super Admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
regexpids	ID/array	Zwraca tylko wyrażenia regularne o podanych identyfikatorach.
selectExpressions	query	Zwraca właściwość `expressions`.

Parametr	Typ	Opis
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `regexpid`, `name`.
countOutput	boolean	Te parametry opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie globalnych wyrażeń regularnych.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "regexp.get",
    "params": {
        "output": ["regexpid","name","description"],
        "selectExpressions": ["expression","expression_type"],
        "regexpids": [1,2],
        "preservekeys": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "1": {
            "regexpid": "1",
            "name": "Systemy plików do wykrywania",
            "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": "Interfejsy sieciowe do wykrywania",
            "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
}
```

Źródło

CRegexp::get() w *ui/include/classes/api/services/CRegexp.php*.

regexp.update

Opis

`object regexp.update(object/array regularExpressions)`

Ta metoda umożliwia aktualizację istniejących globalnych wyrażeń regularnych.

Note:
Ta metoda jest dostępna wyłącznie dla typów użytkowników *Super admin*. Uprawnienia do wywoływania tej metody można cofnąć w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości wyrażenia regularnego do zaktualizowania.

Właściwość `regexpid` musi być zdefiniowana dla każdego obiektu, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz właściwości standardowych metoda akceptuje następujące parametry.

Parametr	Type	Opis
expressions	array	Opcje wyrażeń.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych wyrażeń regularnych we właściwości `regexpids`.

Przykłady

Aktualizacja globalnego wyrażenia regularnego dla wykrywania systemów plików.

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
}
```

Response:

```
{
    "jsonrpc": "2.0",
    "result": {
        "regexpids": [
            "1"
        ]
    },
    "id": 1
}
```

Źródło

CRegexp::update() w *ui/include/classes/api/services/CRegexp.php*.

Wyzwalacz

Ta klasa jest przeznaczona do pracy z wyzwalaczami.

Odwołania do obiektów:

- Wyzwalacz
- Tag wyzwalacza

Dostępne metody:

- trigger.create - tworzenie nowych wyzwalaczy
- trigger.delete - usuwanie wyzwalaczy
- trigger.get - pobieranie wyzwalaczy
- trigger.update - aktualizacja wyzwalaczy

Obiekt wyzwalacza

Następujące obiekty są bezpośrednio powiązane z API `trigger`.

Wyzwalacz

Obiekt wyzwalacza ma następujące właściwości.

Właściwość	Typ	Opis
triggerid	ID	ID wyzwalacza. Zachowanie właściwości: - *tylko do odczytu* - *wymagane* dla operacji aktualizacji
description	string	Nazwa wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
expression	string	Wyrażenie wyzwalacza. Pozycje z ustawionym `value_type` na 5 (binary) lub 6 (JSON) nie mogą być używane w wyrażeniach wyzwalacza. Zachowanie właściwości: - *wymagane* dla operacji tworzenia
event_name	string	Nazwa zdarzenia generowanego przez wyzwalacz.
opdata	string	Dane operacyjne.
comments	string	Dodatkowy opis wyzwalacza.
error	string	Tekst błędu, jeśli wystąpiły jakiekolwiek problemy podczas aktualizacji stanu wyzwalacza. Zachowanie właściwości: - *tylko do odczytu*
flags	integer	Pochodzenie wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* zwykły wyzwalacz; 4 - wyzwalacz przekonwertowany z prototypu. Zachowanie właściwości: - *tylko do odczytu*
lastchange	timestamp	Czas ostatniej zmiany stanu wyzwalacza. Zachowanie właściwości: - *tylko do odczytu*
priority	integer	Poziom ważności wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* niesklasyfikowany; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa.
state	integer	Stan wyrażenia wyzwalacza. Możliwe wartości: 0 - *(domyślnie)* stan wyzwalacza jest aktualny; 1 - bieżący stan wyzwalacza jest nieznany. Zachowanie właściwości: - *tylko do odczytu*
status	integer	Czy wyzwalacz jest włączony czy wyłączony. Możliwe wartości: 0 - *(domyślnie)* włączony; 1 - wyłączony.
templateid	ID	ID nadrzędnego wyzwalacza szablonu. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
type	integer	Czy wyzwalacz może generować wiele zdarzeń problemu. Możliwe wartości: 0 - *(domyślnie)* nie generuj wielu zdarzeń; 1 - generuj wiele zdarzeń.
url	string	URL powiązany z wyzwalaczem.
url_name	string	Etykieta dla URL powiązanego z wyzwalaczem.
value	integer	Czy wyzwalacz jest w stanie OK czy problemu. Możliwe wartości: 0 - *(domyślnie)* OK; 1 - problem. Zachowanie właściwości: - *tylko do odczytu*
recovery_mode	integer	Tryb generowania zdarzenia OK. Możliwe wartości: 0 - *(domyślnie)* Wyrażenie; 1 - Wyrażenie odzyskiwania; 2 - Brak.
recovery_expression	string	Wyrażenie odzyskiwania wyzwalacza. Pozycje z ustawionym `value_type` na 5 (binary) lub 6 (JSON) nie mogą być używane w wyrażeniach odzyskiwania wyzwalacza.
correlation_mode	integer	Zamykanie zdarzeń OK. Możliwe wartości: 0 - *(domyślnie)* Wszystkie problemy; 1 - Wszystkie problemy, jeśli wartości tagów są zgodne.
correlation_tag	string	Tag do dopasowania.
manual_close	integer	Zezwalaj na ręczne zamknięcie. Możliwe wartości: 0 - *(domyślnie)* Nie; 1 - Tak.
uuid	string	Uniwersalny unikalny identyfikator, używany do łączenia importowanych wyzwalaczy z już istniejącymi. Generowany automatycznie, jeśli nie został podany. Zachowanie właściwości: - *obsługiwane* jeśli wyzwalacz należy do szablonu

Tag wyzwalacza

Obiekt tagu wyzwalacza ma następujące właściwości.

Właściwość	Typ	Opis
tag	string	Nazwa tagu wyzwalacza. Zachowanie właściwości: - *wymagane*
value	string	Wartość tagu wyzwalacza. Zachowanie właściwości: - *tylko do odczytu*.

Właściwość	Typ	Opis
object	integer	Typ obiektu, z którego tag został odziedziczony. Możliwe wartości: 0 - szablon; 1 - host; 3 - pozycja. Zachowanie właściwości: - *tylko do odczytu*.
objectid	ID	ID obiektu, z którego tag został odziedziczony. Zachowanie właściwości: - *tylko do odczytu*.
automatic	integer	Typ tagu wyzwalacza: Możliwe wartości: 0 - *(domyślnie)* tag utworzony przez użytkownika; 1 - tag utworzony przez LLD;Zachowanie właściwości: - *tylko do odczytu*.

trigger.create

Opis

`object trigger.create(object/array triggers)`

Ta metoda umożliwia tworzenie nowych wyzwalaczy.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Wyzwalacze do utworzenia.

Oprócz standardowych właściwości wyzwalacza metoda akceptuje następujące parametry.

Parametr	Typ	Opis
dependencies	array	Wyzwalacze, od których zależy wyzwalacz. Wyzwalacze muszą mieć zdefiniowaną tylko właściwość `triggerid`.
tags	array	Tagi wyzwalacza.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych wyzwalaczy we właściwości `triggerids`. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych wyzwalaczy.

Przykłady

Tworzenie wyzwalacza

Utwórz dwa wyzwalacze, z których każdy zależy od jednego innego wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.create",
    "params": [
        {
            "description": "Obciążenie procesora jest zbyt wysokie na {HOST.NAME}",
```

```
            "expression": "last(/Linux server/system.cpu.load[percpu,avg1])>5",
            "dependencies": [
                {
                    "triggerid": "17367"
                }
            ]
        },
        {
            "description": "Usługa została zatrzymana",
            "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17369",
            "17370"
        ]
    },
    "id": 1
}
```

Źródło

CTrigger::create() w *ui/include/classes/api/services/CTrigger.php*.

trigger.delete

Opis

`object trigger.delete(array triggerIds)`

Ta metoda umożliwia usuwanie wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna tylko dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywołania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(array)` ID wyzwalaczy do usunięcia.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory usuniętych wyzwalaczy we właściwości `triggerids`.

Przykłady

Usuń wiele wyzwalaczy

Usuń dwa wyzwalacze.

Żądanie:

```
{
"jsonrpc": "2.0",
"method": "trigger.delete",
"params": [
"12002",
"12003"
],
"id": 1
}
```

Odpowiedź:

```
{
"jsonrpc": "2.0",
"result": {
"triggerids": [
"12002",
"12003"
]
},
"id": 1
}
```

Źródło

CTrigger::delete() w *ui/include/classes/api/services/CTrigger.php*.

trigger.get

Opis

`integer/array trigger.get(object parameters)`

Metoda umożliwia pobieranie wyzwalaczy zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
triggerids	ID/array	Zwraca tylko wyzwalacze o podanych identyfikatorach.
groupids	ID/array	Zwraca tylko wyzwalacze należące do hostów lub szablonów z podanych grup hostów lub grup szablonów.
templateids	ID/array	Zwraca tylko wyzwalacze należące do podanych szablonów.
hostids	ID/array	Zwraca tylko wyzwalacze należące do podanych hostów.
itemids	ID/array	Zwraca tylko wyzwalacze zawierające podane pozycje.
maintenanceids	ID/array	Zwraca tylko wyzwalacze, na które wpływają podane okresy utrzymania.
functions	string/array	Zwraca tylko wyzwalacze używające podanych funkcji. Lista obsługiwanych funkcji znajduje się na stronie obsługiwanych funkcji.

Parametr	Typ	Opis
group	string	Zwraca tylko wyzwalacze należące do hostów lub szablonów z grupy hostów lub grupy szablonów o podanej nazwie.
host	string	Zwraca tylko wyzwalacze należące do hosta o podanej nazwie technicznej.
inherited	boolean	Jeśli ustawiono `true`, zwraca tylko wyzwalacze odziedziczone z szablonu.
templated	boolean	Jeśli ustawiono `true`, zwraca tylko wyzwalacze należące do szablonów.
dependent	boolean	Jeśli ustawiono `true`, zwraca tylko wyzwalacze mające zależności. Jeśli ustawiono `false`, zwraca tylko wyzwalacze niemające zależności.
monitored	flag	Zwraca tylko włączone wyzwalacze należące do monitorowanych hostów i zawierające wyłącznie włączone pozycje.
active	flag	Zwraca tylko włączone wyzwalacze należące do monitorowanych hostów.
maintenance	boolean	Jeśli ustawiono `true`, zwraca tylko włączone wyzwalacze należące do hostów objętych okresem utrzymania.
withUnacknowledgedEvents	flag	Zwraca tylko wyzwalacze mające niepotwierdzone zdarzenia.
withAcknowledgedEvents	flag	Zwraca tylko wyzwalacze, których wszystkie zdarzenia zostały potwierdzone.
withLastEventUnacknowledged	flag	Zwraca tylko wyzwalacze, których ostatnie zdarzenie nie zostało potwierdzone.
skipDependent	flag	Pomija wyzwalacze w stanie problemu, które zależą od innych wyzwalaczy. Należy pamiętać, że inne wyzwalacze są ignorowane, jeśli są wyłączone, mają wyłączone pozycje lub należą do hostów z wyłączonymi pozycjami.
lastChangeSince	timestamp	Zwraca tylko wyzwalacze, które zmieniły stan po podanym czasie.
lastChangeTill	timestamp	Zwraca tylko wyzwalacze, które zmieniły stan przed podanym czasem.
only_true	flag	Zwraca tylko wyzwalacze, które niedawno znajdowały się w stanie problemu (w obrębie `ok_period`).
min_severity	integer	Zwraca tylko wyzwalacze o ważności większej lub równej podanej ważności.
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwraca tylko wyzwalacze z podanymi tagami. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie wyzwalacze. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
expandComment	flag	Rozwija makra w opisie wyzwalacza.
expandDescription	flag	Rozwija makra w nazwie wyzwalacza.
expandExpression	flag	Rozwija funkcje i makra w wyrażeniu wyzwalacza.
selectHostGroups	query	Zwraca grupy hostów, do których należy wyzwalacz, we właściwości `hostgroups`.
selectHosts	query	Zwraca hosty, do których należy wyzwalacz, we właściwości `hosts`.
selectItems	query	Zwraca pozycje zawarte w wyzwalaczu we właściwości `items`.

Parametr	Typ	Opis
selectFunctions	query	Zwraca funkcje używane w wyzwalaczu we właściwości `functions`. Obiekty funkcji reprezentują funkcje używane w wyrażeniu wyzwalacza i mają następujące właściwości: `functionid` - `(ID)` identyfikator funkcji; `itemid` - `(ID)` identyfikator pozycji użytej w funkcji; `function` - `(string)` nazwa funkcji; `parameter` - `(string)` parametr przekazany do funkcji. Parametr zapytania jest zastępowany symbolem $ w zwracanym ciągu.
selectDependencies	query	Zwraca wyzwalacze, od których zależy dany wyzwalacz, we właściwości `dependencies`.
selectDiscoveryData	query	Zwraca właściwość `discoveryData` zawierającą dane obiektu wykrywania wyzwalacza. Obiekt wykrywania wyzwalacza łączy wykryty wyzwalacz z prototypem wyzwalacza, na podstawie którego został wykryty. Ma następujące właściwości: `parent_triggerid` - `(ID)` identyfikator prototypu wyzwalacza, na podstawie którego utworzono wyzwalacz; `status` - `(int)` stan wykrywania wyzwalacza: 0 - *(domyślnie)* wyzwalacz został wykryty, 1 - wyzwalacz nie jest już wykrywany; `ts_delete` - `(timestamp)` czas, po którym wyzwalacz, który nie jest już wykrywany, zostanie usunięty; `ts_disable` - `(timestamp)` czas, po którym wyzwalacz, który nie jest już wykrywany, zostanie wyłączony; `disable_source` - `(int)` wskaźnik określający, czy wyzwalacz został wyłączony przez regułę LLD, czy ręcznie: 0 - *(domyślnie)* wyłączony automatycznie, 1 - wyłączony przez regułę LLD.
selectDiscoveryRule	query	Zwraca regułę wykrywania niskopoziomowego, która utworzyła wyzwalacz, we właściwości `discoveryRule`.
selectLastEvent	query	Zwraca ostatnie istotne zdarzenie wyzwalacza we właściwości `lastEvent`.
selectTags	query	Zwraca tagi wyzwalacza we właściwości `tags`.
inheritedTags	boolean	Zwraca wyzwalacze, które mają podane `tags` również w szablonie/hoście/połączonych szablonach. Możliwe wartości: `true` - szablon/host/połączone szablony również muszą mieć podane tagi; `false` - *(domyślnie)* tagi z szablonu/hosta/połączonych szablonów są ignorowane.
selectInheritedTags	query	Zwraca właściwość `inheritedTags` zawierającą tagi odziedziczone z szablonów, połączonych szablonów i hostów oraz z pozycji, do których odwołują się wyrażenia problemu lub odzyskiwania wyzwalacza.
selectTemplateGroups	query	Zwraca grupy szablonów, do których należy wyzwalacz, we właściwości `templategroups`.
filter	object	Zwraca tylko wyniki dokładnie pasujące do podanego filtra. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami pojedyncza wartość lub tablica wartości, z którymi należy porównać wyniki. Nie obsługuje właściwości typu danych `text` data type. Obsługuje dodatkowe właściwości: `host` - nazwa techniczna hosta, do którego należy wyzwalacz; `hostid` - identyfikator hosta, do którego należy wyzwalacz.

Parametr	Typ	Opis
limitSelects	integer	Ogranicza liczbę rekordów zwracanych przez podzapytania. Dotyczy następujących podzapytań: `selectHosts` - wyniki zostaną posortowane według `host`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `triggerid`, `description`, `status`, `priority`, `lastchange`, `hostname`.
countOutput	boolean	Parametry te opisano w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	
selectTriggerDiscovery	query	Zwraca obiekt wykrywania wyzwalacza we właściwości `triggerDiscovery`. Obiekty wykrywania wyzwalaczy łączą wyzwalacz z prototypem wyzwalacza, na podstawie którego został utworzony. To zapytanie jest **przestarzałe**. Zamiast niego należy użyć `selectDiscoveryData`.

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobieranie danych według ID wyzwalacza

Pobierz wszystkie dane oraz funkcje używane w wyzwalaczu „14062”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "triggerids": "14062",
        "output": "extend",
        "selectFunctions": "extend"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobieranie wyzwalaczy w stanie problemu

Pobierz identyfikator, nazwę i ważność wszystkich wyzwalaczy w stanie problemu i posortuj je według ważności malejąco.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description",
            "priority"
        ],
        "filter": {
            "value": 1
        },
        "sortfield": "priority",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "triggerid": "13907",
            "description": "Procesy samomonitorowania Zabbix zajęte w mniej niż 100%",
            "priority": "4"
        },
```

```
        {
            "triggerid": "13824",
            "description": "Procesy discoverera Zabbix zajęte w więcej niż 75%",
            "priority": "3"
        }
    ],
    "id": 1
}
```

Pobieranie określonego wyzwalacza z tagami

Pobierz określony wyzwalacz z tagami.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ],
        "selectTags": "extend",
        "triggerids": [
            "17578"
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Reguła wykrywania
- Pozycja
- Host
- Grupa hostów
- Grupa szablonów

Źródło

CTrigger::get() w *ui/include/classes/api/services/CTrigger.php*.

trigger.update

Opis

`object trigger.update(object/array triggers)`

Ta metoda umożliwia aktualizację istniejących wyzwalaczy.

> **Note:**
> Ta metoda jest dostępna wyłącznie dla typów użytkowników *Admin* i *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Właściwości wyzwalacza do zaktualizowania.

Właściwość `triggerid` musi być zdefiniowana dla każdego wyzwalacza, wszystkie pozostałe właściwości są opcjonalne. Zaktualizowane zostaną tylko przekazane właściwości, wszystkie pozostałe pozostaną bez zmian.

Oprócz standardowych właściwości wyzwalacza metoda akceptuje następujące parametry.

Parametr	Type	Opis
dependencies	array	Wyzwalacze, od których zależy wyzwalacz. W wyzwalaczach musi być zdefiniowana tylko właściwość `triggerid`.
tags	array	Tagi wyzwalacza.

Wartości zwracane

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych wyzwalaczy we właściwości `triggerids`.

Przykłady

Włączanie wyzwalacza

Włącz wyzwalacz, czyli ustaw jego status na „0”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "13938",
        "status": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Zastępowanie tagów wyzwalacza

Zastąp tagi dla wyzwalacza.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "13938"
        ]
    },
    "id": 1
}
```

Zastępowanie zależności

Zastąp zależności dla wyzwalacza.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "trigger.update",
    "params": {
        "triggerid": "22713",
        "dependencies": [
            {
                "triggerid": "22712"
            },
            {
                "triggerid": "22772"
            }
        ]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "22713"
        ]
    },
    "id": 1
}
```

Źródło

CTrigger::update() w *ui/include/classes/api/services/CTrigger.php*.

Węzeł wysokiej dostępności

Ta klasa została zaprojektowana do pracy z węzłami serwera, które są częścią klastra wysokiej dostępności lub samodzielnej instancji serwera.

Odwołania do obiektów:

- Węzeł wysokiej dostępności

Dostępne metody:

- hanode.get - pobieranie węzłów

Obiekt węzła wysokiej dostępności

Poniższy obiekt jest powiązany z działaniem klastra wysokiej dostępności serwerów Zabbix.

Węzeł wysokiej dostępności

Note:
Węzły są tworzone przez serwer Zabbix i nie mogą być modyfikowane za pomocą API.

Obiekt węzła wysokiej dostępności ma następujące właściwości.

Property	Type	Description
ha_nodeid	ID	ID węzła.
name	string	Nazwa przypisana do węzła, używająca wpisu konfiguracyjnego HANodeName w `zabbix_server.conf`. Puste dla serwera działającego w trybie autonomicznym.
address	string	Adres IP lub nazwa DNS, z której węzeł się łączy.
port	integer	Port, na którym działa węzeł.
lastaccess	integer	Czas heartbeat, czyli czas ostatniej aktualizacji z węzła. Znacznik czasu UTC.
status	integer	Stan węzła. Możliwe wartości: 0 - standby; 1 - zatrzymany ręcznie; 2 - niedostępny; 3 - aktywny.

hanode.get

Opis

`integer/array hanode.get(object parameters)`

Ta metoda umożliwia pobranie listy węzłów klastra wysokiej dostępności zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna tylko dla typów użytkowników *Super admin*. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
ha_nodeids	ID/array	Zwraca tylko węzły o podanych identyfikatorach węzłów.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Obsługiwane właściwości: `name`, `address`, `status`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `name`, `lastaccess`, `status`.
countOutput	flag	Te parametry opisano w komentarzu referencyjnym.
limit	integer	
output	query	
preservekeys	boolean	
sortorder	string/array	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli został użyty parametr `countOutput`.

Przykłady

Pobierz listę węzłów uporządkowaną według statusu

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "preservekeys": true,
        "sortfield": "status",
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
  "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
}
```

Pobierz listę określonych węzłów według ich identyfikatorów

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "ha_nodeids": ["ckuo7i1nw000e0sajwfttc1mp", "ckuo7i1nv000c0sajz85xcrtt"]
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobierz listę zatrzymanych węzłów

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["ha_nodeid", "address", "port"],
        "filter": {
```

```
            "status": 1
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "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
}
```

Pobierz liczbę węzłów zapasowych

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "countOutput": true,
        "filter": {
            "status": 0
        }
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "3",
    "id": 1
}
```

Sprawdzenie statusu węzłów pod określonymi adresami IP

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "hanode.get",
    "params": {
        "output": ["name", "status"],
        "filter": {
            "address": ["192.168.1.7", "192.168.1.13"]
        }
    },
```

```
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "name": "node3",
            "status": "0"
        },
        {
            "name": "node-active",
            "status": "3"
        }
    ],
    "id": 1
}
```

Źródło

CHaNode::get() w *ui/include/classes/api/services/CHaNode.php*.

Zadanie

Ta klasa jest przeznaczona do pracy z zadaniami (takimi jak sprawdzanie pozycji lub reguł wykrywania niskiego poziomu bez przeładowania konfiguracji).

Odwołania do obiektów:

- Task
  - Obiekt żądania „Wykonaj teraz”
  - Obiekt żądania „Odśwież konfigurację proxy”
  - Obiekt żądania „Informacje diagnostyczne”
    * Obiekt żądania statystyk
    * Obiekt wyniku statystyk

Dostępne metody:

- task.create - utwórz nowe zadania
- task.get - pobierz zadania

Obiekt task

Następujące obiekty są bezpośrednio powiązane z API `task`.

Task

Obiekt task ma następujące właściwości:

Property	Type	Description
taskid	ID	ID zadania. Zachowanie właściwości: *- tylko do odczytu*

Property	Type	Description
type	integer	Typ zadania. Możliwe wartości: 1 - Informacje diagnostyczne; 2 - Odśwież konfigurację proxy; 6 - Wykonaj teraz. Użytkownicy typu *Admin* i *User* mogą tworzyć tylko zadania „Wykonaj teraz”. Zwróć uwagę, że `task.get` zawsze zwraca „7” (Podsumowanie wykonania zadania). Zachowanie właściwości: - *wymagane*
status	integer	Status zadania. Możliwe wartości: 1 - nowe zadanie; 2 - zadanie w toku; 3 - zadanie ukończone; 4 - zadanie wygasło. Zachowanie właściwości: - *tylko do odczytu*
clock	timestamp	Czas utworzenia zadania. Zachowanie właściwości: - *tylko do odczytu*
ttl	integer	Czas w sekundach, po którym zadanie wygasa. Zachowanie właściwości: - *tylko do odczytu*
proxyid	ID	ID proxy, dla którego zbierane są statystyki informacji diagnostycznych. Zachowanie właściwości: - *obsługiwane*, jeśli `type` jest ustawione na „Informacje diagnostyczne” lub „Odśwież konfigurację proxy”
request	object	Obiekt żądania zadania zgodnie z typem zadania: Obiekt zadania „Wykonaj teraz” jest opisany szczegółowo poniżej; Obiekt zadania „Odśwież konfigurację proxy” jest opisany szczegółowo poniżej; Obiekt zadania „Informacje diagnostyczne” jest opisany szczegółowo poniżej. Zachowanie właściwości: - *wymagane*
result	object	Obiekt wyniku zadania informacji diagnostycznych. Może zawierać NULL, jeśli wynik nie jest jeszcze gotowy. Obiekt wyniku jest opisany szczegółowo poniżej. Zachowanie właściwości: - *tylko do odczytu*

Obiekt żądania „Execute now”

Obiekt żądania zadania „Execute now” ma następujące właściwości.

Właściwość	Typ	Opis
itemid	ID	ID pozycji i reguł wykrywania niskiego poziomu. Użytkownicy typu *Admin* i *User* mogą wykonać „Execute now” dla pozycji na hostach, do których mają uprawnienie odczytu i zapisu (*read-write*) albo uprawnienie *read* oraz włączoną akcję `invoke_execute_now` action dla swojej roli. To samo dotyczy użytkowników typu *Admin* w przypadku reguł wykrywania niskiego poziomu (LLD).

Obiekt żądania „Refresh proxy configuration”

Obiekt żądania zadania „Refresh proxy configuration” ma następujące właściwości.

Właściwość	Type	Opis
proxyids	array	ID proxy.

Obiekt żądania „Diagnostic information”

Obiekt żądania zadania informacji diagnostycznych ma następujące właściwości. Obiekt żądania statystyk dla wszystkich typów właściwości jest opisany szczegółowo poniżej.

Właściwość	Typ	Opis
historycache	object	Żądanie statystyk pamięci podręcznej historii. Dostępne na serwer i proxy.
valuecache	object	Żądanie statystyk pamięci podręcznej pozycji. Dostępne na serwer.
preprocessing	object	Żądanie statystyk menedżera przetwarzania wstępnego. Dostępne na serwer i proxy.
alerting	object	Żądanie statystyk menedżera alertów. Dostępne na serwer.
lld	object	Żądanie statystyk menedżera LLD. Dostępne na serwer.

Obiekt żądania statystyk

Obiekt żądania statystyk służy do określenia, jaki typ informacji powinien być zbierany o wewnętrznych procesach serwera/proxy. Ma on następujące właściwości.

Właściwość	Typ	Opis
stats	query	Właściwości obiektu statystyk, które mają zostać zwrócone. Lista dostępnych pól dla każdego typu statystyk informacji diagnostycznych jest opisana szczegółowo poniżej. Domyślnie: `extend` zwróci wszystkie dostępne pola statystyk.
top	object	Obiekt służący do sortowania i ograniczania zwracanych wartości statystyk. Lista dostępnych pól dla każdego typu statystyk informacji diagnostycznych jest opisana szczegółowo poniżej. Przykład: { “source.alerts”: 10 }

Lista pól statystyk dostępnych dla każdego typu żądania informacji diagnostycznych

Dla każdej właściwości typu żądania informacji diagnostycznych można żądać następujących pól statystyk.

Typ diagnostyczny	Dostępne pola	Opis
historycache	items	Liczba elementów w pamięci podręcznej.
	values	Liczba wartości w pamięci podręcznej.

Typ diagnostyczny	Dostępne pola	Opis
	memory	Statystyki pamięci współdzielonej (wolne miejsce, liczba używanych fragmentów, liczba wolnych fragmentów, maksymalny rozmiar wolnego fragmentu).
	memory.data	Statystyki pamięci współdzielonej pamięci podręcznej danych historii.
	memory.index	Statystyki pamięci współdzielonej pamięci podręcznej indeksu historii.
valuecache	items	Liczba elementów w pamięci podręcznej.
	values	Liczba wartości w pamięci podręcznej.
	memory	Statystyki pamięci współdzielonej (wolne miejsce, liczba używanych fragmentów, liczba wolnych fragmentów, maksymalny rozmiar wolnego fragmentu).
	mode	Tryb pamięci podręcznej wartości.
preprocessing	values	Liczba wartości w kolejce.
	preproc.values	Liczba wartości w kolejce z krokami przetwarzania wstępnego.
alerting	alerts	Liczba alertów w kolejce.
lld	rules	Liczba reguł w kolejce.
	values	Liczba wartości w kolejce.

Lista pól sortowania dostępnych dla każdego typu żądania informacji diagnostycznych

Następujące pola statystyczne mogą być używane do sortowania i ograniczania żądanych informacji.

Typ diagnostyczny	Dostępne pola	Typ
historycache	values	integer
valuecache	values	integer
	request.values	integer
preprocessing	values	integer
alerting	media.alerts	integer
	source.alerts	integer
lld	values	integer

Obiekt wyniku statystyk

Obiekt wyniku statystyk jest pobierany w polu `result` obiektu zadania.

Właściwość	Typ	Opis
status	integer	Status wyniku zadania. Możliwe wartości: -1 - podczas wykonywania zadania wystąpił błąd; 0 - wynik zadania został utworzony. Zachowanie właściwości: *- tylko do odczytu*
data	string/object	Wyniki zgodne z obiektem żądania statystyk dla określonego zadania informacji diagnostycznych. Zawiera ciąg komunikatu o błędzie, jeśli podczas wykonywania zadania wystąpił błąd.

task.create

Opis

`object task.create(object/array tasks)`

Ta metoda umożliwia tworzenie zadań.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Zadanie do utworzenia.

Metoda akceptuje zadania ze standardowymi właściwościami zadania.

Należy pamiętać, że zadania „Wykonaj teraz” można tworzyć tylko dla następujących typów pozycji/reguł wykrywania:

- agent Zabbix (pasywny)
- Proste sprawdzenie
- agent SNMP (v1/v2/v3)
- Wewnętrzny Zabbix
- Sprawdzenie zewnętrzne
- Monitor bazy danych
- agent HTTP
- agent IPMI
- agent SSH
- agent TELNET
- agent JMX
- Obliczana
- Zależna pozycja
- Skrypt
- Przeglądarka

Jeśli pozycja/reguła wykrywania jest typu „Zależna pozycja”, to jej pozycja nadrzędna również musi być jednym z powyższych typów.

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory utworzonych zadań we właściwości `taskids`. Jedno zadanie jest tworzone dla każdej pozycji i reguły wykrywania niskiego poziomu. Kolejność zwróconych identyfikatorów odpowiada kolejności przekazanych `itemids`.

Przykłady

Tworzenie zadania

Utwórz zadanie „Wykonaj teraz” dla pozycji i reguły wykrywania niskiego poziomu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 6,
            "request": {
                "itemid": "10092"
            }
        },
        {
            "type": 6,
            "request": {
                "itemid": "10093"
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1",
            "2"
        ]
```

```
    },
    "id": 1
}
```

Utwórz zadanie „Odśwież konfigurację proxy” dla dwóch proxy.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.create",
    "params": [
        {
            "type": 2,
            "request": {
                "proxyids": ["10459", "10460"]
            }
        }
    ],
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "1"
        ]
    },
    "id": 1
}
```

Utwórz zadanie „Informacje diagnostyczne”.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "taskids": [
            "3"
        ]
    },
    "id": 1
}
```

Zobacz także

- Task
- Obiekt żądania „Wykonaj teraz”
- Obiekt żądania „Odśwież konfigurację proxy”
- Obiekt żądania „Informacje diagnostyczne”

Źródło

CTask::create() w *ui/include/classes/api/services/CTask.php*.

task.get

Opis

`integer/array task.get(object parameters)`

Metoda umożliwia pobieranie zadań zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna wyłącznie dla typu użytkownika *Super admin*. Uprawnienia do wywoływania tej metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
taskids	ID/array	Zwraca tylko zadania o podanych identyfikatorach.
output	query	Te parametry są opisane w komentarzu referencyjnym.
preservekeys	boolean	

Zwrócone wartości

`(integer/array)` Zwraca tablicę obiektów.

Przykłady

Pobierz zadanie według ID

Pobierz wszystkie dane dotyczące zadania o ID „1”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "task.get",
    "params": {
        "output": "extend",
        "taskids": "1"
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "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
}
```


Zobacz także

- Task
- Obiekt żądania „Wykonaj teraz”
- Obiekt żądania „Odśwież konfigurację proxy”
- Obiekt żądania „Informacje diagnostyczne”

Źródło

CTask::get() w *ui/include/classes/api/services/CTask.php*.

Zdarzenie

Ta klasa jest przeznaczona do pracy ze zdarzeniami.

Odwołania do obiektów:

- Event
- Tag zdarzenia
- URL typu nośnika

Dostępne metody:

- event.get - pobierz zdarzenia
- event.acknowledge - potwierdź zdarzenia

Obiekt zdarzenia

Następujące obiekty są bezpośrednio powiązane z API `event`.

Zdarzenie

Note:
Zdarzenia są tworzone przez serwer Zabbix i nie mogą być modyfikowane przez API.

Obiekt zdarzenia ma następujące właściwości.

Właściwość	Typ	Opis
eventid	ID	ID zdarzenia.
source	integer	Typ zdarzenia. Możliwe wartości: 0 - zdarzenie utworzone przez wyzwalacz; 1 - zdarzenie utworzone przez regułę wykrywania; 2 - zdarzenie utworzone przez aktywną autorejestrację agenta; 3 - zdarzenie wewnętrzne; 4 - zdarzenie utworzone przy aktualizacji stanu usługi.
object	integer	Typ obiektu powiązanego ze zdarzeniem. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz”: 0 - wyzwalacz. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez regułę wykrywania”: 1 - wykryty host; 2 - wykryta usługa. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez aktywną autorejestrację agenta”: 3 - host z autorejestracji. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie wewnętrzne”: 0 - wyzwalacz; 4 - pozycja; 5 - reguła LLD. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przy aktualizacji stanu usługi”: 6 - usługa.
objectid	ID	ID powiązanego obiektu.
acknowledged	integer	Czy zdarzenie zostało potwierdzone.
clock	timestamp	Czas utworzenia zdarzenia.
ns	integer	Nanosekundy w momencie utworzenia zdarzenia.

Właściwość	Typ	Opis
name	string	Rozwinięta nazwa zdarzenia.
value	integer	Stan powiązanego obiektu. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz” lub „zdarzenie utworzone przy aktualizacji stanu usługi”: 0 - OK; 1 - problem. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie utworzone przez regułę wykrywania”: 0 - host lub usługa działa; 1 - host lub usługa nie działa; 2 - host lub usługa wykryte; 3 - host lub usługa utracone. Możliwe wartości, jeśli `source` jest ustawione na „zdarzenie wewnętrzne”: 0 - stan „normalny”; 1 - stan „nieznany” lub „niewspierany”. Zachowanie właściwości: - *obsługiwane* jeśli `source` jest ustawione na „zdarzenie utworzone przez wyzwalacz”, „zdarzenie utworzone przez regułę wykrywania”, „zdarzenie wewnętrzne” lub „zdarzenie utworzone przy aktualizacji stanu usługi”
severity	integer	Bieżący poziom ważności zdarzenia. Możliwe wartości: 0 - niesklasyfikowane; 1 - informacja; 2 - ostrzeżenie; 3 - średni; 4 - wysoki; 5 - katastrofa.
r_eventid	ID	ID zdarzenia odzyskania.
c_eventid	ID	ID zdarzenia użytego do nadpisania (zamknięcia) bieżącego zdarzenia zgodnie z globalną regułą korelacji. Zobacz `correlationid`, aby zidentyfikować dokładną regułę korelacji. Ten parametr jest zdefiniowany tylko wtedy, gdy zdarzenie zostało zamknięte przez globalną regułę korelacji.
cause_eventid	ID	ID zdarzenia przyczyny.
correlationid	ID	ID reguły korelacji, która spowodowała zamknięcie problemu. Ten parametr jest zdefiniowany tylko wtedy, gdy zdarzenie zostało zamknięte przez globalną regułę korelacji.
userid	ID	ID użytkownika, który zamknął zdarzenie (jeśli zdarzenie zostało zamknięte ręcznie).
suppressed	integer	Czy zdarzenie jest wyciszone. Możliwe wartości: 0 - zdarzenie jest w stanie normalnym; 1 - zdarzenie jest wyciszone.
opdata	string	Dane operacyjne z rozwiniętymi makrami.
urls	array	Aktywne adresy URL typu nośnika.

Znacznik zdarzenia

Obiekt znacznika zdarzenia ma następujące właściwości.

Właściwość	Type	Opis
tag	string	Nazwa znacznika zdarzenia.
value	string	Wartość znacznika zdarzenia.

URL typu nośnika

Obiekt URL typu nośnika ma następujące właściwości.

Właściwość	Typ	Opis
name	string	Nazwa URL zdefiniowanego dla typu nośnika.
url	string	Wartość URL zdefiniowana dla typu nośnika.

Wyniki będą zawierać wpisy tylko dla aktywnych typów nośników z włączonym wpisem menu zdarzenia. Makra użyte we właściwościach zostaną rozwinięte, ale jeśli jedna z właściwości zawiera nierozwinięte makro, obie właściwości zostaną wykluczone z wyników. Obsługiwane makra opisano w sekcji *Obsługiwane makra*.

event.acknowledge

Opis

`object event.acknowledge(object/array parameters)`

Ta metoda umożliwia aktualizację zdarzeń. Można wykonać następujące akcje aktualizacji:

- Zamknij zdarzenie. Jeśli zdarzenie jest już rozwiązane, ta akcja zostanie pominięta.
- Potwierdź zdarzenie. Jeśli zdarzenie jest już potwierdzone, ta akcja zostanie pominięta.
- Cofnij potwierdzenie zdarzenia. Jeśli zdarzenie nie jest potwierdzone, ta akcja zostanie pominięta.
- Dodaj wiadomość.
- Zmień powagę zdarzenia. Jeśli zdarzenie ma już taką samą powagę, ta akcja zostanie pominięta.
- Wycisz zdarzenie. Jeśli zdarzenie jest już wyciszone, ta akcja zostanie pominięta.
- Cofnij wyciszenie zdarzenia. Jeśli zdarzenie nie jest wyciszone, ta akcja zostanie pominięta.
- Zmień rangę zdarzenia na przyczynę. Jeśli zdarzenie ma już rangę przyczyny, ta akcja zostanie pominięta.
- Zmień rangę zdarzenia na objaw. Jeśli zdarzenie ma już rangę objawu, ta akcja zostanie pominięta.

Attention:
Można aktualizować tylko zdarzenia wyzwalaczy.
 Można aktualizować tylko zdarzenia problemów.
 Do zamknięcia zdarzenia lub zmiany jego powagi wymagane są uprawnienia odczytu/zapisu dla wyzwalacza.
 Aby zamknąć zdarzenie, w wyzwalaczu musi być dozwolone ręczne zamknięcie.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object/array)` Parametry zawierające identyfikatory zdarzeń i operacje aktualizacji, które powinny zostać wykonane.

Parameter	Type	Description
eventids	ID/array	Identyfikatory zdarzeń, które należy potwierdzić. Zachowanie parametru: - *wymagane*

Parameter	Type	Description
action	integer	Operacja(e) aktualizacji zdarzenia. Możliwe wartości bitowe: 1 - zamknięcie problemu; 2 - potwierdzenie zdarzenia; 4 - dodanie komunikatu; 8 - zmiana ważności; 16 - anulowanie potwierdzenia zdarzenia; 32 - wyciszenie zdarzenia; 64 - anulowanie wyciszenia zdarzenia; 128 - zmiana rangi zdarzenia na przyczynę; 256 - zmiana rangi zdarzenia na objaw. To pole jest maską bitową; dopuszczalna jest dowolna suma możliwych wartości bitowych (na przykład 6 w celu potwierdzenia zdarzenia i pozostawienia komunikatu albo 34 w celu potwierdzenia i wyciszenia zdarzenia). Zachowanie parametru: - *wymagane*
cause_eventid	ID	Identyfikator zdarzenia będącego przyczyną. Zachowanie parametru: - *wymagane*, jeśli `action` zawiera bit „zmiana rangi zdarzenia na objaw”
message	string	Treść komunikatu. Zachowanie parametru: - *wymagane*, jeśli `action` zawiera bit „dodanie komunikatu”
severity	integer	Nowa ważność zdarzeń. Możliwe wartości: 0 - niesklasyfikowane; 1 - informacje; 2 - ostrzeżenie; 3 - średnia; 4 - wysoka; 5 - katastrofalna. Zachowanie parametru: - *wymagane*, jeśli `action` zawiera bit „zmiana ważności”
suppress_until	integer	Znacznik czasu Unix, do którego zdarzenie musi być wyciszone. Jeśli ustawiono wartość „0”, wyciszenie będzie obowiązywać bezterminowo. Zachowanie parametru: - *wymagane*, jeśli `action` zawiera bit „wyciszenie zdarzenia”

Zwracane wartości

`(object)` Zwraca obiekt zawierający identyfikatory zaktualizowanych zdarzeń w właściwości `eventids`.

Przykłady

Potwierdzanie zdarzenia

Potwierdź pojedyncze zdarzenie i pozostaw wiadomość.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
```

```
    "params": {
        "eventids": "20427",
        "action": 6,
        "message": "Problem rozwiązany."
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427"
        ]
    },
    "id": 1
}
```

Zmiana ważności zdarzenia

Zmień ważność wielu zdarzeń i pozostaw wiadomość.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": ["20427", "20428"],
        "action": 12,
        "message": "Maintenance required to fix it.",
        "severity": 4
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            "20427",
            "20428"
        ]
    },
    "id": 1
}
```

Wyciszanie zdarzenia bezterminowo

Wycisz pojedyncze zdarzenie bez czasu wygaśnięcia.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 32,
        "suppress_until": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427
        ]
    },
    "id": 1
}
```

Cofnięcie wyciszenia zdarzenia

Cofnij wyciszenie pojedynczego wcześniej wyciszonego zdarzenia.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": "20427",
        "action": 64
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427
        ]
    },
    "id": 1
}
```

Potwierdź i wycisz zdarzenie

Potwierdź jedno lub więcej zdarzeń, dodaj wiadomość i wycisz je w tym samym żądaniu.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.acknowledge",
    "params": {
        "eventids": [
            "20427",
            "20428"
        ],
        "action": 38,
        "message": "Suppressing during planned maintenance.",
        "suppress_until": 0
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": {
        "eventids": [
            20427,
            20428
```

```
        ]
    },
    "id": 1
}
```

Źródło

CEvent::acknowledge() w *ui/include/classes/api/services/CEvent.php*.

event.get

Opis

`integer/array event.get(object parameters)`

Metoda umożliwia pobieranie zdarzeń zgodnie z podanymi parametrami.

Attention:
Ta metoda może zwrócić zdarzenia usuniętej encji, jeśli nie zostały one jeszcze usunięte przez housekeepera.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywołania metody można odebrać w ustawieniach roli użytkownika. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące oczekiwany wynik.

Metoda obsługuje następujące parametry.

Parameter	Type	Description
eventids	ID/array	Zwraca tylko zdarzenia o podanych ID.
groupids	ID/array	Zwraca tylko zdarzenia utworzone przez obiekty należące do podanych grup hostów.
hostids	ID/array	Zwraca tylko zdarzenia utworzone przez obiekty należące do podanych hostów.
objectids	ID/array	Zwraca tylko zdarzenia utworzone przez podane obiekty.
source	integer	Zwraca tylko zdarzenia o podanym typie. Listę obsługiwanych typów zdarzeń znajdziesz na stronie obiektu zdarzenia. Domyślnie: 0 - zdarzenia wyzwalacza.
object	integer	Zwraca tylko zdarzenia utworzone przez obiekty o podanym typie. Listę obsługiwanych typów obiektów znajdziesz na stronie obiektu zdarzenia. Domyślnie: 0 - wyzwalacz.
acknowledged	boolean	Jeśli ustawione na `true`, zwraca tylko potwierdzone zdarzenia.
action	integer	Zwraca tylko zdarzenia, dla których wykonano podane akcje aktualizacji zdarzeń. W przypadku wielu akcji użyj sumy dowolnych akceptowalnych wartości bitmapy jako maski bitowej (na przykład 34 dla potwierdzenia i wyciszenia zdarzenia).
action_userids	ID/array	Zwraca tylko zdarzenia o podanych ID użytkowników, którzy wykonali akcje aktualizacji zdarzeń.
suppressed	boolean	Jeśli ustawione na `true`, zwraca tylko wyciszone zdarzenia.
symptom	boolean	Jeśli ustawione na `true`, zwraca tylko zdarzenia objawowe.
severities	integer/array	Zwraca tylko zdarzenia o podanych poziomach ważności zdarzeń. Dotyczy tylko wtedy, gdy obiektem jest wyzwalacz.
trigger_severities	integer/array	Zwraca tylko zdarzenia o podanych poziomach ważności wyzwalacza. Dotyczy tylko wtedy, gdy obiektem jest wyzwalacz.

Parameter	Type	Description
evaltype	integer	Metoda oceny tagów. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
tags	array	Zwraca tylko zdarzenia o podanych tagach. Format: `[{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie zdarzenia. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Równa się; 2 - Nie zawiera; 3 - Nie równa się; 4 - Istnieje; 5 - Nie istnieje.
eventid_from	string	Zwraca tylko zdarzenia o ID większym lub równym podanemu ID.
eventid_till	string	Zwraca tylko zdarzenia o ID mniejszym lub równym podanemu ID.
time_from	timestamp	Zwraca tylko zdarzenia utworzone w podanym czasie lub później.
time_till	timestamp	Zwraca tylko zdarzenia utworzone w podanym czasie lub wcześniej.
problem_time_from	timestamp	Zwraca tylko zdarzenia, które znajdowały się w stanie problemu od `problem_time_from`, niezależnie od ich bieżącego statusu. Dotyczy tylko wtedy, gdy źródłem jest zdarzenie wyzwalacza, a obiektem jest wyzwalacz. Ten parametr jest wymagany, jeśli określono `problem_time_till`.
problem_time_till	timestamp	Zwraca tylko zdarzenia, które znajdowały się w stanie problemu do `problem_time_till`, niezależnie od ich bieżącego statusu. Dotyczy tylko wtedy, gdy źródłem jest zdarzenie wyzwalacza, a obiektem jest wyzwalacz. Ten parametr jest wymagany, jeśli określono `problem_time_from`.
value	integer/array	Zwraca tylko zdarzenia o podanych wartościach.
selectAcknowledges	query	Zwraca właściwość `acknowledges` z aktualizacjami zdarzeń. Aktualizacje zdarzeń są sortowane w kolejności odwrotnej chronologicznie. Obiekt aktualizacji zdarzenia ma następujące właściwości: `acknowledgeid` - `(ID)` ID potwierdzenia; `userid` - `(ID)` ID użytkownika, który zaktualizował zdarzenie; `clock` - `(timestamp)` czas aktualizacji zdarzenia; `message` - `(string)` treść wiadomości; `action` - `(integer)` wykonana akcja aktualizacji, zobacz `event.acknowledge`; `old_severity` - `(integer)` ważność zdarzenia przed tą akcją aktualizacji; `new_severity` - `(integer)` ważność zdarzenia po tej akcji aktualizacji; `suppress_until` - `(timestamp)` czas, do którego zdarzenie będzie wyciszone; `taskid` - `(ID)` ID zadania, jeśli bieżące zdarzenie przechodzi zmianę rangi; `maintenanceid` - `(ID)` ID utrzymania; `username` - `(string)` nazwa użytkownika, który zaktualizował zdarzenie; `name` - `(string)` imię użytkownika, który zaktualizował zdarzenie; `surname` - `(string)` nazwisko użytkownika, który zaktualizował zdarzenie. Obsługuje `count`.
selectAlerts	query	Zwraca właściwość `alerts` z alertami wygenerowanymi przez zdarzenie. Alerty są sortowane w kolejności odwrotnej chronologicznie.

Parameter	Type	Description
selectHosts	query	Zwraca właściwość `hosts` z hostami zawierającymi obiekt, który utworzył zdarzenie. Obsługiwane tylko dla zdarzeń generowanych przez wyzwalacze, pozycje lub reguły LLD.
selectRelatedObject	query	Zwraca właściwość `relatedObject` z obiektem, który utworzył zdarzenie. Typ zwracanego obiektu zależy od typu zdarzenia.
selectSuppressionData	query	Zwraca właściwość `suppression_data` z listą aktywnych utrzymań i ręcznych wyciszeń: `maintenanceid` - `(ID)` ID utrzymania; `userid` - `(ID)` ID użytkownika, który wyciszył zdarzenie; `suppress_until` - `(integer)` czas, do którego zdarzenie jest wyciszone.
selectTags	query	Zwraca właściwość `tags` z tagami zdarzenia.
filter	object	Zwraca tylko te wyniki, które dokładnie pasują do podanego filtra. Przyjmuje obiekt, w którym klucze są nazwami właściwości, a wartościami jest pojedyncza wartość lub tablica wartości, z którymi ma nastąpić dopasowanie. Nie obsługuje właściwości typu `text` data type.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `eventid`, `objectid`, `clock`. Możliwe wartości przy użyciu razem z `groupBy`: `objectid`. Możliwe wartości przy użyciu razem z `countOutput` i `groupBy`: `objectid`, `rowscount`.
groupBy	string/array	Grupuje wyniki według podanych właściwości. Określone właściwości zostaną zwrócone w wynikach. Możliwe wartości: `objectid`.
countOutput	boolean	Te parametry są opisane w komentarzu referencyjnym.
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`, ale nie użyto parametru `groupBy`
- tablicę obiektów z wynikami agregacji, jeśli użyto parametru `groupBy`

Przykłady

Pobieranie zdarzeń wyzwalacza

Pobierz najnowsze zdarzenia z wyzwalacza ”22395”.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
```

```
        "selectAcknowledges": "extend",
        "selectSuppressionData": "extend",
        "selectTags": "extend",
        "objectids": "22395",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20",
            "source": "0",
            "object": "0",
            "objectid": "22395",
            "clock": "1728658089",
            "value": "0",
            "acknowledged": "0",
            "ns": "461809482",
            "name": "Load average is too high (per CPU load over 1.5 for 5m)",
            "severity": "0",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "suppressed": "0",
            "cause_eventid": "0",
            "acknowledges": [],
            "opdata": "Load averages(1m 5m 15m): (1.6328 3.0522 2.2515), # of 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": "Load average is too high (per CPU load over 1.5 for 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": "Load averages(1m 5m 15m): (1.6328 3.0522 2.2515), # of 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
}
```

Pobieranie zdarzeń według przedziału czasu

Pobierz wszystkie zdarzenia utworzone między 17 a 18 października 2012 roku, w odwrotnej kolejności chronologicznej.

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "time_from": "1350432000",
        "time_till": "1350518400",
        "sortfield": ["clock", "eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": [
        {
            "eventid": "20617",
            "source": "0",
            "object": "0",
            "objectid": "14282",
            "clock": "1350477816",
            "value": "1",
            "acknowledged": "0",
            "ns": "0",
            "name": "Mniej niż 25% wolnego miejsca w pamięci podręcznej historii",
            "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": "Procesy Zabbix trapper są zajęte w ponad 75%",
            "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": "Procesy Zabbix trapper są zajęte w ponad 75%",
            "severity": "3",
            "r_eventid": "20616",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "cause_eventid": "0",
            "opdata": "",
            "suppressed": "0",
            "urls": []
        }
    ],
    "id": 1
}
```


Pobieranie zdarzeń potwierdzonych przez określonego użytkownika

Pobieranie zdarzeń potwierdzonych przez użytkownika o ID=10

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "event.get",
    "params": {
        "output": "extend",
        "action": 2,
        "action_userids": [10],
        "selectAcknowledges": ["userid", "action"],
        "sortfield": ["eventid"],
        "sortorder": "DESC"
    },
    "id": 1
}
```

Odpowiedź:


```
{
    "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
}
```

Pobieranie najważniejszych wyzwalaczy według liczby zdarzeń problemowych

Pobierz 5 najważniejszych wyzwalaczy o poziomach ważności „Warning”, „Average”, „High” lub „Disaster” wraz z liczbą zdarzeń problemowych w określonym przedziale czasu.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Alert
- Pozycja
- Host
- Reguła LLD
- Usługa
- Wyzwalacz

Źródło

CEvent::get() w *ui/include/classes/api/services/CEvent.php*.

Ślad APM

Ta klasa jest przeznaczona do pracy ze śladami zbieranymi na potrzeby monitorowania wydajności aplikacji (APM).

Odwołania do obiektów:

- Ślad APM
- Zdarzenie
- Łącze

Dostępne metody:

- apmtrace.get - pobieranie śladów

Obiekt śladu APM

Poniższe obiekty są bezpośrednio powiązane z API `apmtrace`.

Note:
Każda właściwość, z wyjątkiem `span_count` i `error_count`, odpowiada kolumnie w tabeli `otel_traces` źródła danych telemetrycznych. Większość właściwości odpowiada kolumnie o tej samej nazwie (np. właściwość `span_name` odpowiada kolumnie `SpanName`). Jeśli nazwy się różnią, kolumny źródła danych określono w poniższych tabelach. Listę kolumn można znaleźć w sekcji Obsługiwane kolumny danych telemetrycznych.

Ślad APM

Obiekt śladu APM reprezentuje ślad (ścieżkę, którą żądanie przebywa przez aplikacje i usługi). Ślad składa się z co najmniej jednego zakresu; każdy zakres reprezentuje pojedynczą operację.

Obiekt śladu APM jest oparty na głównym zakresie śladu (zakresie, który rozpoczyna ślad i nie ma zakresu nadrzędnego). Wszystkie właściwości zawierają dane głównego zakresu, z wyjątkiem `span_count` i `error_count`, które są obliczane na podstawie wszystkich zakresów śladu.

Obiekt śladu APM ma następujące właściwości:

Właściwość	Typ	Opis
timestamp	integer	Czas uniksowy (w nanosekundach), w którym rozpoczął się główny zakres. Zachowanie właściwości: - *tylko do odczytu*
traceid	string	Identyfikator śladu współdzielony przez wszystkie zakresy należące do śladu. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
spanid	string	Identyfikator głównego zakresu. Zachowanie właściwości: - *tylko do odczytu*
trace_state	string	Dodatkowe dane, jeśli istnieją, dołączane do śladu przez narzędzia śledzenia (zobacz W3C Trace Context). Zachowanie właściwości: - *tylko do odczytu*
span_name	string	Nazwa operacji reprezentowanej przez główny zakres (np. `POST /zabbix.php`). Zachowanie właściwości: - *tylko do odczytu*
span_kind	string	Rola głównego zakresu. Możliwe wartości: `Unspecified` - rodzaj zakresu nie został określony; `Internal` - operacja wykonana wewnątrz aplikacji; `Server` - obsługa przychodzącego żądania; `Client` - żądanie wychodzące; `Producer` - utworzenie operacji asynchronicznej (np. wysłanie wiadomości); `Consumer` - przetwarzanie operacji asynchronicznej. Zachowanie właściwości: - *tylko do odczytu*
service_name	string	Nazwa usługi, która utworzyła główny zakres (np. `zabbix-nginx`), pobrana z atrybutu zasobu OpenTelemetry `service.name`. Zachowanie właściwości: - *tylko do odczytu*
resource_attributes	object	Atrybuty zasobu klucz-wartość głównego zakresu (np. `host.name: eabf8c2a5ddc`). Zachowanie właściwości: - *tylko do odczytu*
scope_name	string	Nazwa zakresu instrumentacji, który utworzył główny zakres. Zachowanie właściwości: - *tylko do odczytu*
scope_version	string	Wersja zakresu instrumentacji. Zachowanie właściwości: - *tylko do odczytu*
span_attributes	object	Atrybuty zakresu klucz-wartość głównego zakresu (np. `http.method: POST`). Zachowanie właściwości: - *tylko do odczytu*
duration	integer	Czas trwania śladu w nanosekundach. Zachowanie właściwości: - *tylko do odczytu*

Właściwość	Typ	Opis
status_code	string	Stan operacji reprezentowanej przez główny zakres. Możliwe wartości: `Unset` - głównemu zakresowi nie przypisano jawnego stanu; `Ok` - operacja została ukończona pomyślnie; `Error` - operacja została ukończona z błędem. Zachowanie właściwości: - *tylko do odczytu*
status_message	string	Dodatkowe informacje, jeśli istnieją, dotyczące stanu głównego zakresu. Zachowanie właściwości: - *tylko do odczytu*
events	array	Tablica obiektów zdarzeń głównego zakresu, czyli istotnych momentów podczas operacji (np. wyjątków). Zachowanie właściwości: - *tylko do odczytu*
links	array	Tablica obiektów łączy głównego zakresu wskazujących powiązane zakresy w tym samym lub innym śladzie. Zachowanie właściwości: - *tylko do odczytu*
span_count	integer	Łączna liczba zakresów w śladzie. Zachowanie właściwości: - *tylko do odczytu*
error_count	integer	Liczba zakresów w śladzie, dla których `status_code` ma wartość `Error`. Zachowanie właściwości: - *tylko do odczytu*

Zdarzenie

Obiekt zdarzenia ma następujące właściwości.

Właściwość	Typ	Opis
timestamp	integer	Czas uniksowy (w nanosekundach), w którym wystąpiło zdarzenie. Kolumna źródła danych: `Events.Timestamp`
name	string	Nazwa zdarzenia. Kolumna źródła danych: `Events.Name`
attributes	object	Atrybuty klucz-wartość zawierające dodatkowe informacje o zdarzeniu, jeśli są dostępne. Kolumna źródła danych: `Events.Attributes`

Link

Obiekt linku ma następujące właściwości.

Property	Type	Description
traceid	string	ID śladu zawierającego połączony span. Kolumna źródła danych: `Links.TraceId`
spanid	string	ID połączonego spanu. Wraz z `traceid` jednoznacznie identyfikuje połączony span. Kolumna źródła danych: `Links.SpanId`

Property	Type	Description
trace_state	string	Stan śladu, jeśli istnieje, połączonego spanu (zobacz W3C Trace Context). Kolumna źródła danych: `Links.TraceState`
attributes	object	Atrybuty klucz-wartość zawierające dodatkowe informacje o linku. Kolumna źródła danych: `Links.Attributes`

apmtrace.get

Opis

`integer/array apmtrace.get(object parameters)`

Metoda umożliwia pobieranie śladów zgodnie z podanymi parametrami.

Note:
Ta metoda jest dostępna dla użytkowników dowolnego typu. Uprawnienia do wywoływania metody można odebrać w ustawieniach ról użytkowników. Więcej informacji można znaleźć w sekcji Role użytkowników.

Parametry

`(object)` Parametry definiujące żądane dane wyjściowe.

Metoda obsługuje następujące parametry.

Parametr	Typ	Opis
time_from	timestamp	Zwraca tylko ślady, których główny span rozpoczął się o podanym czasie lub później. Zachowanie parametru: - *wymagany*
time_till	timestamp	Zwraca tylko ślady, których główny span rozpoczął się przed podanym czasem. Zachowanie parametru: - *wymagany*
traceids	string/array	Zwraca tylko ślady o podanych identyfikatorach.
resource_attributes	object/array	Zwraca tylko ślady, których główny span ma podane atrybuty zasobu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie ślady. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
resource_attributes_evaltype	integer	Metoda oceny warunków w `resource_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.

Parametr	Typ	Opis
span_attributes	object/array	Zwraca tylko ślady, których główny span ma podane atrybuty spanu. Format: `[{"key": "<key>", "value": "<value>", "operator": "<operator>"}, ...]`. Pusta tablica zwraca wszystkie ślady. Możliwe wartości operatora: 0 - *(domyślnie)* Zawiera; 1 - Jest równe; 2 - Nie zawiera; 3 - Nie jest równe; 4 - Istnieje; 5 - Nie istnieje.
span_attributes_evaltype	integer	Metoda oceny warunków w `span_attributes`. Możliwe wartości: 0 - *(domyślnie)* And/Or; 2 - Or.
min_duration	integer	Zwraca tylko ślady, których główny span ma wartość `duration` większą lub równą podanej wartości, w nanosekundach.
max_duration	integer	Zwraca tylko ślady, których główny span ma wartość `duration` mniejszą od podanej wartości, w nanosekundach.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są pojedyncze wartości lub tablice wartości, z którymi należy porównać wyniki. Obsługiwane właściwości: `traceid`, `spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `status_code`, `status_message`.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez uwzględniania wielkości liter). Przyjmuje obiekt, w którym kluczami są nazwy właściwości, a wartościami są wyszukiwane ciągi znaków. Jeśli nie podano dodatkowych opcji, wyszukiwanie będzie wykonywane za pomocą `LIKE "%…%"`. Obsługiwane właściwości: `trace_state`, `span_name`, `service_name`, `scope_name`, `scope_version`, `status_message`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Możliwe wartości: `timestamp`, `traceid`, `spanid`, `trace_state`, `span_name`, `span_kind`, `service_name`, `scope_name`, `scope_version`, `duration`, `status_code`, `status_message`.
offset	integer	Liczba rekordów pomijanych przed zwróceniem wyniku.
countOutput	boolean	Parametry te opisano w komentarzu referencyjnym.
excludeSearch	boolean	
limit	integer	
output	query	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Zwracane wartości

`(integer/array)` Zwraca:

- tablicę obiektów;
- liczbę pobranych obiektów, jeśli użyto parametru `countOutput`.

Przykłady

Pobieranie najwolniejszych śladów

Pobierz dwa najwolniejsze ślady usługi `zabbix-nginx`. Wyszukaj ślady z okresu od 10:00 do 11:00 1 października 2026 r. (UTC) i uwzględnij tylko te, których czas trwania wynosił co najmniej 1 sekundę.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Zobacz także

- Span APM
- Metryka APM
- Dziennik APM

Źródło

CApmTrace::get() w *ui/include/classes/api/services/CApmTrace.php*.

Notacja Sygnatura metody

Każda strona metody API zawiera sygnaturę metody.

Przykład: `object action.create(object/array actions)`

Sygnatura	Opis
`object`	Typ danych zwracanych przez metodę.
`action.create`	Pełna nazwa metody.
`object/array`	Typ danych akceptowanych przez metodę.
`actions`	Krótkie podsumowanie tego, co akceptuje metoda.

Typy danych

API Zabbix obsługuje następujące typy danych jako dane wejściowe:

Type	Description
ID	Unikalny identyfikator używany do odwoływania się do encji.
boolean	Wartość logiczna (`true` lub `false`).
flag	Wartość uznawana za `true`, jeśli została przekazana i nie jest równa `null`; w przeciwnym razie wartość jest uznawana za `false`.
integer	Liczba całkowita.
float	Liczba zmiennoprzecinkowa.
string	Ciąg tekstowy.
text	Dłuższy ciąg tekstowy.
timestamp	Znacznik czasu Unix.
array	Uporządkowana sekwencja wartości (zwykła tablica).
object	Tablica asocjacyjna.
query	Wartość definiująca dane, które mają zostać zwrócone. Wartość może być zdefiniowana jako tablica nazw właściwości (aby zwrócić tylko określone właściwości) lub jako jedna ze wstępnie zdefiniowanych wartości: `extend` - zwraca wszystkie właściwości obiektu; `count` - zwraca liczbę pobranych rekordów, obsługiwane tylko przez niektóre podwybory.

Attention:
API Zabbix zawsze zwraca wartości wyłącznie jako ciągi znaków lub tablice.

Zachowanie właściwości

Niektóre właściwości obiektu są oznaczone krótkimi etykietami opisującymi ich zachowanie. Używane są następujące etykiety:

- ***tylko do odczytu*** - wartość właściwości jest ustawiana automatycznie i nie może być definiowana ani zmieniana przez użytkownika, nawet w niektórych szczególnych warunkach (np. *tylko do odczytu* dla obiektów dziedziczonych lub obiektów wykrytych);
- ***tylko do zapisu*** - wartość właściwości może zostać ustawiona, ale później nie można uzyskać do niej dostępu;
- ***stała*** - wartość właściwości może zostać ustawiona podczas tworzenia obiektu, ale później nie może być zmieniona;
- ***obsługiwana*** - ustawienie wartości właściwości nie jest wymagane, ale jest dozwolone w niektórych szczególnych warunkach (np. *obsługiwana*, jeśli `type` jest ustawione na ”Simple check”, ”External check”, ”SSH agent”, ”TELNET agent” lub ”HTTP agent”); należy jednak pamiętać, że właściwości oznaczone jako *obsługiwana* mogą nadal być ustawiane na swoje wartości domyślne niezależnie od warunków;
- ***wymagana*** - ustawienie wartości właściwości jest wymagane dla wszystkich operacji (z wyjątkiem operacji pobierania) lub w niektórych szczególnych warunkach (np. *wymagana* dla operacji tworzenia; *wymagana*, jeśli `operationtype` jest ustawione na ”global script”, a `opcommand_hst` nie jest ustawione).

Note:
W przypadku operacji aktualizacji właściwość jest uznawana za „ustawioną”, gdy zostanie ustawiona podczas operacji aktualizacji.

Właściwości, które nie są oznaczone etykietami, są opcjonalne.

Zachowanie parametrów

Niektóre parametry operacji są oznaczone krótkimi etykietami opisującymi ich zachowanie dla danej operacji. Używane są następujące etykiety:

- ***tylko do odczytu*** - wartość parametru jest ustawiana automatycznie i nie może być definiowana ani zmieniana przez użytkownika, nawet w niektórych szczególnych warunkach (np. *tylko do odczytu* dla obiektów dziedziczonych lub obiektów wykrytych);
- ***tylko do zapisu*** - wartość parametru może zostać ustawiona, ale później nie można uzyskać do niej dostępu;
- ***obsługiwany*** - ustawienie wartości parametru nie jest wymagane, ale jest dozwolone w niektórych szczególnych warunkach (np. *obsługiwany*, jeśli `operating_mode` obiektu Proxy jest ustawiony na "passive proxy"); należy jednak pamiętać, że parametry oznaczone jako *obsługiwany* nadal mogą być ustawiane na wartości domyślne niezależnie od warunków;
- ***wymagany*** - wartość parametru musi zostać ustawiona.

Parametry, które nie są oznaczone etykietami, są opcjonalne.

Zastrzeżona wartość ID "0" Zastrzeżona wartość ID "0" może być używana do filtrowania elementów i usuwania obiektów , do których istnieją odniesienia. Na przykład, aby usunąć odwołanie do proxy z hosta , proxyid powinno być ustawione na 0 ("proxyid": "0") lub aby filtrować hosty monitorowane przez opcję serwera proxyids powinno być ustawione na 0 ("proxyids": "0").

Wspólne parametry metod "get" Następujące parametry są obsługiwane przez wszystkie metody `get`:

Parametr	Typ	Opis
countOutput	boolean	Zwraca liczbę rekordów w wyniku zamiast rzeczywistych danych.
editable	boolean	Jeśli ustawiono na `true`, zwraca tylko obiekty, do których użytkownik ma uprawnienia zapisu. Domyślnie: `false`.
excludeSearch	boolean	Zwraca wyniki, które nie pasują do kryteriów podanych w parametrze `search`.
filter	object	Zwraca tylko te wyniki, które dokładnie odpowiadają podanemu filtrowi. Akceptuje obiekt, w którym kluczami są nazwy właściwości (np. właściwości obiektu Host w `host.get`, właściwości obiektu Item w `item.get` itd.), a wartościami są pojedyncza wartość lub tablica wartości do dopasowania. Nie obsługuje właściwości typu danych `text` data type. Należy pamiętać, że niektóre metody mają specyficzną funkcjonalność dla tego parametru, opisaną na stronie danej metody (np. parametr `filter` w host.get obsługuje również właściwości interfejsu hosta).
limit	integer	Ogranicza liczbę zwracanych rekordów.
output	query	Właściwości obiektu, które mają zostać zwrócone. Należy pamiętać, że identyfikator obiektu (tj. `hostid`, `itemid` itd.) jest zawsze uwzględniany w odpowiedzi, nawet jeśli nie został określony w parametrze `output`. Domyślnie: `extend`.
preservekeys	boolean	Używa identyfikatorów jako kluczy w wynikowej tablicy.
search	object	Zwraca wyniki pasujące do podanego wzorca (bez rozróżniania wielkości liter). Akceptuje obiekt, w którym kluczami są nazwy właściwości (np. właściwości obiektu Host w `host.get`, właściwości obiektu Item w `item.get` itd.), a wartościami są ciągi znaków do wyszukania. Jeśli nie podano dodatkowych opcji, zostanie wykonane wyszukiwanie `LIKE "%...%"`. Obsługuje tylko właściwości typu danych `string` i `text` data type. Należy pamiętać, że niektóre metody mają specyficzną funkcjonalność dla tego parametru, opisaną na stronie danej metody (np. parametr `search` w host.get obsługuje również właściwości interfejsu hosta).

Parametr	Typ	Opis
searchByAny	boolean	Jeśli ustawiono na `true`, zwraca wyniki pasujące do dowolnego z kryteriów podanych w parametrze `filter` lub `search`, zamiast do wszystkich. Domyślnie: `false`.
searchWildcardsEnabled	boolean	Jeśli ustawiono na `true`, włącza użycie znaku "*" jako znaku wieloznacznego w parametrze `search`. Domyślnie: `false`.
sortfield	string/array	Sortuje wynik według podanych właściwości. Listę właściwości, które mogą być używane do sortowania, można znaleźć w opisie konkretnej metody API get. Makra nie są rozwijane przed sortowaniem. Jeśli nie podano wartości, dane zostaną zwrócone bez sortowania.
sortorder	string/array	Kolejność sortowania. Jeśli przekazano tablicę, każda wartość zostanie dopasowana do odpowiadającej jej właściwości podanej w parametrze `sortfield`. Możliwe wartości: `ASC` - *(domyślnie)* rosnąco; `DESC` - malejąco.
startSearch	boolean	Parametr `search` będzie porównywał początek pól, czyli zamiast tego wykona wyszukiwanie `LIKE "…%"`. Ignorowane, jeśli `searchWildcardsEnabled` jest ustawione na `true`.

Flagi pochodzenia encji Metody get zwracają właściwość `flags` dla encji powiązanych z wykrywaniem niskopoziomowym (reguła LLD/prototyp reguły LLD, pozycja/prototyp pozycji itd.). Ta właściwość jest przydatna do określenia, czy encja została wykryta, czy nie, ponieważ możliwości edycji wykrytych encji są ograniczone.

Właściwość `flags` zwraca wynik oparty na kombinacji (operacja „+") tych wartości:

Value	Description
0	Encja bazowa (pozycja, wyzwalacz, wykres, host)
1	Reguła wykrywania niskopoziomowego
2	Dowolny prototyp (prototyp pozycji, prototyp wyzwalacza, prototyp reguły LLD itd.)
4	Wykryta encja (wykryta pozycja, wyzwalacz, wykres, host, reguła LLD)

Łączna wartość zwracana przez właściwość `flags` może być następująca:

Value	Combination of	Description
0	0	Zwykła encja (pozycja, wyzwalacz, wykres, host).
2	2	Prototyp encji (prototyp pozycji, prototyp wyzwalacza itd.).
6	2+4	Wykryta pozycja, wyzwalacz, wykres, host (utworzone z prototypu).
1	1	Reguła wykrywania niskopoziomowego.
3	1+2	Prototyp reguły wykrywania niskopoziomowego.
5	1+4	Wykryta reguła wykrywania niskopoziomowego (utworzona z prototypu).
7	1+2+4	Wykryty prototyp reguły wykrywania niskopoziomowego.

Przykłady Sprawdzenie uprawnień użytkownika

Czy użytkownik ma uprawnienia do zapisu do hostów, których nazwy zaczynają się od "MySQL" lub "Linux" ?

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
        "search": {
            "host": ["MySQL", "Linux"]
```

```
        },
        "editable": true,
        "startSearch": true,
        "searchByAny": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "0",
    "id": 1
}
```

Note:
Wynik zero oznacza brak hostów z uprawnieniami do odczytu/zapisu.

Zliczanie niedopasowań

Policz liczbę hostów, których nazwy nie zawierają podciągu ”ubuntu”

Żądanie:

```
{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "countOutput": true,
        "search": {
            "host": "ubuntu"
        },
        "excludeSearch": true
    },
    "id": 1
}
```

Odpowiedź:

```
{
    "jsonrpc": "2.0",
    "result": "44",
    "id": 1
}
```

Wyszukiwanie hostów przy użyciu symboli wieloznacznych

Znajdź hosty, których nazwa zawiera słowo „server” i które mają porty interfejsów „10050” lub „10071”. Posortuj wynik według nazwy hosta malejąco i ogranicz go do 5 hostów.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Wyszukiwanie hostów przy użyciu symboli wieloznacznych z parametrem ”preservekeys”

Jeśli dodasz parametr ”preservekeys” do poprzedniego żądania, wynik zostanie zwrócony jako tablica asocjacyjna, gdzie kluczami

są identyfikatory obiektów.

Żądanie:

```
{
    "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
}
```

Odpowiedź:

```
{
    "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
}
```

Załącznik 2. Zmiany od 7.4 do 8.0

Zmiany powodujące niezgodność

Inne zmiany i poprawki błędów

Załącznik 3. Zmiany w wersji 8.0

8.0.1 Ta wersja nie została jeszcze wydana.

Rozszerzenia

Przegląd Chociaż Zabbix oferuje wiele funkcji, zawsze istnieje miejsce na dodatkową funkcjonalność. Rozszerzenia są wygodnym sposobem modyfikowania i zwiększania możliwości monitorowania w Zabbix bez zmiany jego kodu źródłowego.

Możesz rozszerzyć funkcjonalność Zabbix, korzystając albo z wbudowanych opcji rozszerzeń (pozycje trapper, parametry użytkownika itp.), albo używając lub tworząc własne rozszerzenia (moduły ładowalne, wtyczki itp.).

Ta sekcja zawiera przegląd wraz z odwołaniami do wszystkich opcji rozszerzania Zabbix.

Zbieranie danych za pomocą niestandardowych poleceń Pozycje trapper

Pozycje trapper to pozycje, które przyjmują dane przychodzące zamiast pobierać je w zapytaniu. Pozycje trapper są przydatne do wysyłania określonych danych do serwera Zabbix lub proxy, na przykład okresowych danych o dostępności i wydajności w przypadku długo działających skryptów użytkownika. Wysyłanie danych do serwera Zabbix lub proxy jest możliwe za pomocą narzędzia Zabbix sender lub protokołu Zabbix sender. Wysyłanie danych do serwera Zabbix jest również możliwe za pomocą metody API `history.push`.

Kontrole zewnętrzne

Kontrola zewnętrzna to pozycja służąca do wykonywania kontroli przez uruchomienie pliku wykonywalnego, na przykład skryptu powłoki lub pliku binarnego.

Kontrole zewnętrzne są wykonywane przez serwer Zabbix lub proxy (gdy host jest monitorowany przez proxy) i nie wymagają uruchomionego agenta na monitorowanym hoście.

Parametry użytkownika

Parametr użytkownika to polecenie zdefiniowane przez użytkownika (powiązane z kluczem zdefiniowanym przez użytkownika), które po wykonaniu może pobrać dane potrzebne z hosta, na którym działa agent Zabbix. Parametry użytkownika są przydatne do konfigurowania pozycji agenta lub agenta 2, które nie są predefiniowane w Zabbix.

Pozycje agenta Zabbix `system.run[]`

Pozycja agenta Zabbix `system.run[]` służy do wykonywania polecenia zdefiniowanego przez użytkownika (powiązanego ze wstępnie zdefiniowanym kluczem `system.run[]`, na przykład `system.run[myscript.sh]`) na hoście, na którym działa agent Zabbix.

Note:
Pozycje `system.run[]` są domyślnie wyłączone. Jeśli są używane, należy je włączyć (zezwolić na ich użycie) i zdefiniować w pliku konfiguracyjnym agenta Zabbix lub agenta Zabbix 2 (parametr konfiguracji `AllowKey` lub `AllowKeyRegexp`).

Attention:
Polecenia zdefiniowane przez użytkownika w pozycjach, takich jak sprawdzenia zewnętrzne, parametry użytkownika i pozycje agenta Zabbix `system.run[]`, są wykonywane przez użytkownika systemu operacyjnego używanego do uruchamiania komponentów Zabbix. Aby wykonać te polecenia, ten użytkownik musi mieć niezbędne uprawnienia.

Pozycje HTTP agent

Pozycja HTTP agent służy do wykonywania żądań danych za pośrednictwem HTTP/HTTPS. Pozycje HTTP agent są przydatne do wysyłania żądań do punktów końcowych HTTP w celu pobierania danych z usług takich jak *Elasticsearch* i *OpenWeatherMap*, sprawdzania stanu interfejsu API Zabbix lub stanu serwera WWW Apache albo NGINX itd. Pozycje HTTP agent (z włączonym przechwytywaniem) mogą również działać jako pozycje trapper.

Pozycje skryptowe

Pozycja skryptowa to pozycja służąca do wykonywania zdefiniowanego przez użytkownika kodu JavaScript, który pobiera dane przez HTTP/HTTPS. Pozycje skryptowe są przydatne, gdy funkcjonalność zapewniana przez pozycje agenta HTTP nie jest wystarczająca. Na przykład w wymagających scenariuszach zbierania danych, które wymagają wielu kroków lub złożonej logiki, pozycję skryptową można skonfigurować tak, aby wykonała wywołanie HTTP, następnie przetworzyła otrzymane dane, a potem przekazała przekształconą wartość do drugiego wywołania HTTP.

Note:
Pozycje HTTP agent oraz pozycje skryptowe są obsługiwane przez serwer Zabbix i proxy i nie wymagają uruchomionego agenta na monitorowanym hoście.

Zaawansowane rozszerzenia Moduły ładowalne

Moduły ładowalne, napisane w C, są wszechstronną i wydajnościowo zoptymalizowaną opcją rozszerzania funkcjonalności komponentów Zabbix (serwer, proxy, agent) na platformach UNIX. Moduł ładowalny jest zasadniczo biblioteką współdzieloną używaną przez demon Zabbix i ładowaną podczas uruchamiania. Biblioteka powinna zawierać określone funkcje, aby proces Zabbix mógł wykryć, że plik jest rzeczywiście modułem, który może załadować i z którym może pracować.

Moduły ładowalne mają szereg zalet, w tym możliwość dodawania nowych metryk lub implementowania dowolnej innej logiki (na przykład eksportu danych historii Zabbix), wysoką wydajność oraz możliwość tworzenia, używania i udostępniania zapewnianej przez nie funkcjonalności. Przyczynia się to do bezproblemowego utrzymania i pomaga łatwiej dostarczać nową funkcjonalność niezależnie od bazowego kodu Zabbix.

Moduły ładowalne są szczególnie przydatne w złożonych środowiskach monitorowania. Podczas monitorowania systemów wbudowanych, przy dużej liczbie monitorowanych parametrów lub przy użyciu ciężkich skryptów ze złożoną logiką albo długim czasem uruchamiania, rozszerzenia takie jak parametry użytkownika, pozycje agenta Zabbix `system.run[]` oraz kontrole zewnętrzne będą miały wpływ na wydajność. Moduły ładowalne oferują sposób rozszerzania funkcjonalności Zabbix bez poświęcania wydajności.

Wtyczki

Wtyczki rozszerzają możliwości monitorowania Zabbix agent 2. Napisane w języku programowania Go, stanowią alternatywę dla modułów ładowalnych (napisanych w C).

Możliwe jest opracowanie własnej wtyczki (niezależnego kolektora danych) dla Zabbix agent 2. Zobacz Centrum dewelopera.

Dostosowywanie alertów Webhooki

Webhook to typ mediów w Zabbix, który umożliwia rozszerzenie możliwości alertowania Zabbix na zewnętrzne oprogramowanie, takie jak systemy helpdesk, czaty lub komunikatory. Podobnie jak pozycje skryptowe, webhooki są przydatne do wykonywania wywołań HTTP przy użyciu niestandardowego kodu JavaScript, na przykład do wysyłania powiadomień do różnych platform, takich jak Microsoft Teams, Discord i Jira. Możliwe jest również zwrócenie pewnych danych (na przykład o utworzonych zgłoszeniach helpdesk), które są następnie wyświetlane w Zabbix.

Istniejące webhooki są dostępne w repozytorium Git Zabbix. Informacje dotyczące tworzenia niestandardowych webhooków można znaleźć w dokumencie Wytyczne dotyczące tworzenia webhooków.

Skrypty alertów

Skrypt alertu to typ mediów w Zabbix, który umożliwia utworzenie alternatywnego sposobu (skryptu) obsługi alertów Zabbix. Skrypty alertów są przydatne, jeśli istniejące typy mediów do wysyłania alertów w Zabbix nie spełniają Twoich oczekiwań.

Dostosowanie frontendu Własne motywy

Można zmienić wygląd wizualny frontendu Zabbix, używając własnych motywów. Zobacz instrukcje dotyczące tworzenia i stosowania własnych motywów.

Moduły frontend

Moduły frontend zapewniają możliwość rozszerzenia funkcjonalności frontend Zabbix przez dodawanie modułów innych firm lub tworzenie własnych. Za pomocą modułów frontend można dodawać nowe pozycje menu, odpowiadające im widoki, akcje itp.

Skrypty globalne Skrypt globalny to zdefiniowany przez użytkownika zestaw poleceń, które mogą być wykonywane na monitorowanym celu (przez interpreter powłoki (/bin/sh)), w zależności od skonfigurowanego zakresu i uprawnień użytkownika. Skrypty globalne można skonfigurować dla następujących działań:

- Operacja akcji
- Ręczne działanie na hoście
- Ręczne działanie na zdarzeniu

Skrypty globalne są przydatne w wielu przypadkach. Na przykład, jeśli są skonfigurowane dla operacji akcji lub ręcznych działań na hoście, można używać skryptów globalnych do automatycznego lub ręcznego wykonywania zdalnych poleceń, takich jak ponowne uruchamianie aplikacji (serwera WWW, middleware, CRM itp.) lub zwalnianie miejsca na dysku (usuwanie starszych plików, czyszczenie `/tmp` itp.). Inny przykład: jeśli są skonfigurowane dla ręcznych działań na zdarzeniach, można używać skryptów globalnych do zarządzania zgłoszeniami problemów w systemach zewnętrznych.

Skrypty globalne mogą być wykonywane przez serwer Zabbix, proxy lub agent.

> **Attention:**
> Polecenia zdefiniowane przez użytkownika są wykonywane z poziomu użytkownika systemu operacyjnego używanego do uruchamiania komponentów Zabbix. Aby wykonać te polecenia, ten użytkownik musi mieć niezbędne uprawnienia.

Zabbix API Zabbix API to API oparte na HTTP, będące częścią frontend Zabbix. Za pomocą Zabbix API można wykonywać następujące operacje:

- Programowo pobierać i modyfikować konfigurację Zabbix.
- Importować i eksportować konfigurację Zabbix.
- Uzyskiwać dostęp do danych historycznych i trendów Zabbix.
- Konfigurować aplikacje do współpracy z Zabbix.
- Integrować Zabbix z oprogramowaniem firm trzecich.
- Automatyzować rutynowe zadania.

Zabbix API składa się z wielu metod, które są umownie pogrupowane w oddzielne API. Każda metoda wykonuje określone zadanie. Informacje o dostępnych metodach, a także przegląd funkcji udostępnianych przez Zabbix API, można znaleźć w referencji metod Zabbix API.

Moduły ładowalne

Przegląd

Moduły ładowalne oferują rozwiązanie nastawione na wydajność do rozszerzania funkcjonalności Zabbix.

Funkcjonalność Zabbix można rozszerzać na wiele sposobów, na przykład za pomocą parametrów użytkownika, kontroli zewnętrznych oraz pozycji agenta Zabbix system.run. Rozwiązania te działają bardzo dobrze, ale mają jedną istotną wadę, a mianowicie fork(). Zabbix musi utworzyć nowy proces za każdym razem, gdy obsługuje metrykę użytkownika, co nie jest korzystne dla wydajności. Zwykle nie stanowi to dużego problemu, jednak może być poważnym wyzwaniem podczas monitorowania systemów wbudowanych, przy dużej liczbie monitorowanych parametrów lub w przypadku ciężkich skryptów ze złożoną logiką albo długim czasem uruchamiania.

Obsługa modułów ładowalnych umożliwia rozszerzanie agenta Zabbix, serwera i proxy bez poświęcania wydajności.

Moduł ładowalny jest zasadniczo biblioteką współdzieloną używaną przez demon Zabbix i ładowaną podczas uruchamiania. Biblioteka powinna zawierać określone funkcje, aby proces Zabbix mógł wykryć, że plik jest rzeczywiście modułem, który można załadować i z którym można pracować.

Moduły ładowalne mają szereg zalet. Bardzo ważne są wysoka wydajność i możliwość implementacji dowolnej logiki, ale być może najważniejszą zaletą jest możliwość tworzenia, używania i udostępniania modułów Zabbix. Przyczynia się to do bezproblemowego utrzymania i ułatwia dostarczanie nowej funkcjonalności w prostszy sposób oraz niezależnie od bazowego kodu Zabbix.

Licencjonowanie modułów i ich dystrybucja w postaci binarnej są regulowane licencją AGPL-3.0 (moduły są linkowane z Zabbix w czasie działania i używają nagłówków Zabbix; cały kod Zabbix jest objęty licencją AGPL-3.0 od wersji Zabbix 7.0). Zabbix nie gwarantuje zgodności binarnej.

Stabilność API modułów jest gwarantowana w trakcie jednego cyklu wydania Zabbix LTS (Long Term Support). Stabilność API Zabbix nie jest gwarantowana (technicznie możliwe jest wywoływanie wewnętrznych funkcji Zabbix z modułu, ale nie ma gwarancji, że takie moduły będą działać).

API modułu

Aby biblioteka współdzielona była traktowana jako moduł Zabbix, powinna implementować i eksportować kilka funkcji. Obecnie w API modułu Zabbix istnieje sześć funkcji, z których tylko jedna jest obowiązkowa, a pozostałe pięć jest opcjonalnych.

Obowiązkowy interfejs

Jedyną obowiązkową funkcją jest `zbx_module_api_version()`:

```
int zbx_module_api_version(void);
```

Funkcja ta powinna zwracać wersję API zaimplementowaną przez ten moduł, a aby moduł mógł zostać załadowany, wersja ta musi być zgodna z wersją API modułu obsługiwaną przez Zabbix. Wersja API modułu obsługiwana przez Zabbix to `ZBX_MODULE_API_VERSION`. Dlatego ta funkcja powinna zwracać tę stałą. Stara stała `ZBX_MODULE_API_VERSION_ONE`, używana w tym celu, jest obecnie zdefiniowana jako równa `ZBX_MODULE_API_VERSION`, aby zachować zgodność źródłową, jednak jej użycie nie jest zalecane.

Opcjonalny interfejs

Opcjonalne funkcje to `zbx_module_init()`, `zbx_module_item_list()`, `zbx_module_item_timeout()`, `zbx_module_history_writ` oraz `zbx_module_uninit()`:

```
int zbx_module_init(void);
```

Ta funkcja powinna wykonać niezbędną inicjalizację modułu (jeśli jest wymagana). W przypadku powodzenia powinna zwrócić `ZBX_MODULE_OK`. W przeciwnym razie powinna zwrócić `ZBX_MODULE_FAIL`. W tym drugim przypadku Zabbix nie zostanie uruchomiony.

```
ZBX_METRIC  *zbx_module_item_list(void);
```

Ta funkcja powinna zwrócić listę pozycji obsługiwanych przez moduł. Każda pozycja jest zdefiniowana w strukturze `ZBX_METRIC`; szczegóły znajdują się w sekcji poniżej. Lista jest zakończona strukturą `ZBX_METRIC` z polem "key" ustawionym na NULL.

```
void    zbx_module_item_timeout(int timeout);
```

Jeśli moduł eksportuje `zbx_module_item_list()`, to funkcja ta jest używana przez Zabbix do określenia ustawień limitu czasu w pliku konfiguracyjnym Zabbix, których powinny przestrzegać sprawdzenia pozycji zaimplementowane przez moduł. Tutaj parametr "timeout" jest podawany w sekundach.

```
ZBX_HISTORY_WRITE_CBS   zbx_module_history_write_cbs(void);
```

Ta funkcja powinna zwracać funkcje wywołania zwrotnego, których serwer Zabbix użyje do eksportu historii różnych typów danych. Funkcje wywołania zwrotnego są udostępniane jako pola struktury `ZBX_HISTORY_WRITE_CBS`; pola mogą mieć wartość NULL, jeśli moduł nie jest zainteresowany historią określonego typu.

```
int zbx_module_uninit(void);
```

Ta funkcja powinna wykonać niezbędną deinicjalizację (jeśli jest wymagana), taką jak zwolnienie zaalokowanych zasobów, zamknięcie deskryptorów plików itp.

Wszystkie funkcje są wywoływane raz podczas uruchamiania Zabbix, gdy moduł jest ładowany, z wyjątkiem `zbx_module_uninit()`, która jest wywoływana raz podczas zamykania Zabbix, gdy moduł jest rozładowywany.

Definiowanie pozycji

Każda pozycja jest definiowana w strukturze `ZBX_METRIC`:

```
typedef struct
{
    char        *key;
    unsigned    flags;
    int     (*function)();
    char        *test_param;
}
ZBX_METRIC;
```

Tutaj **key** to klucz pozycji (np. ”dummy.random”), **flags** to albo CF_HAVEPARAMS, albo 0 (w zależności od tego, czy pozycja przyjmuje parametry, czy nie), **function** to funkcja C implementująca pozycję (np. zbx_module_dummy_random), a **test_param** to lista parametrów używana, gdy agent Zabbix jest uruchamiany z flagą –p (np. ”1,1000”, może mieć wartość NULL). Przykładowa definicja może wyglądać tak:

```
static ZBX_METRIC keys[] =
{
    { "dummy.random", CF_HAVEPARAMS, zbx_module_dummy_random, "1,1000" },
    { NULL }
}
```

Każda funkcja implementująca pozycję powinna przyjmować dwa parametry wskaźnikowe: pierwszy typu AGENT_REQUEST, a drugi typu 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;
}
```

Funkcje te powinny zwracać SYSINFO_RET_OK, jeśli wartość pozycji została pomyślnie pobrana. W przeciwnym razie powinny zwracać SYSINFO_RET_FAIL. Zobacz przykład modułu ”dummy” poniżej, aby uzyskać szczegółowe informacje o tym, jak pobierać informacje z AGENT_REQUEST i jak ustawiać informacje w AGENT_RESULT.

Udostępnianie callbacków eksportu historii

Attention:
Eksport historii za pomocą modułu nie jest już obsługiwany przez proxy Zabbix.

Moduł może określać funkcje do eksportu danych historii według typu: Numeric (float), Numeric (unsigned), Character, Text i 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;
```

Każda z nich powinna przyjmować jako argumenty tablicę history o history_num elementach. W zależności od typu danych historii przeznaczonych do eksportu, history jest odpowiednio tablicą następujących struktur:

```
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;
```

Callbacki będą używane przez procesy synchronizacji historii serwera Zabbix na końcu procedury synchronizacji historii, po zapisaniu danych do bazy danych Zabbix i zapisaniu ich w pamięci podręcznej wartości.

Attention:
W przypadku wewnętrznego błędu w module eksportu historii zaleca się, aby moduł był napisany w taki sposób, by nie blokował całego monitoringu do czasu odzyskania sprawności, lecz zamiast tego odrzucał dane i pozwalał serwerowi Zabbix kontynuować działanie.

Budowanie modułów

Moduły są obecnie przeznaczone do budowania wewnątrz drzewa źródeł Zabbix, ponieważ API modułu zależy od niektórych struktur danych zdefiniowanych w nagłówkach Zabbix.

Najważniejszym nagłówkiem dla modułów ładowanych dynamicznie jest `include/module.h`, który definiuje te struktury danych. Inne niezbędne nagłówki systemowe, które pomagają `include/module.h` działać poprawnie, to `stdlib.h` i `stdint.h`.

Mając te informacje na uwadze, wszystko jest gotowe do zbudowania modułu. Moduł powinien zawierać `stdlib.h`, `stdint.h` i `module.h`, a skrypt budowania powinien upewnić się, że pliki te znajdują się w ścieżce dołączania. Szczegóły można znaleźć w przykładzie modułu "dummy" poniżej.

Innym przydatnym nagłówkiem jest `include/zbxcommon.h`, który definiuje funkcję `zabbix_log()`, mogącą służyć do logowania i debugowania.

Parametry konfiguracji

Zabbix agent, serwer i proxy obsługują dwa parametry służące do obsługi modułów:

- LoadModulePath - pełna ścieżka do lokalizacji modułów ładowalnych
- LoadModule - moduł lub moduły ładowane podczas uruchamiania.

Moduły muszą znajdować się w katalogu określonym przez LoadModulePath albo ścieżka musi poprzedzać nazwę modułu. Jeśli poprzedzająca ścieżka jest bezwzględna (zaczyna się od /), LoadModulePath jest ignorowany. Dozwolone jest użycie wielu parametrów LoadModule.

Na przykład, aby rozszerzyć Zabbix agent, można dodać następujące parametry:

```
LoadModulePath=/usr/local/lib/zabbix/agent/
LoadModule=mariadb.so
LoadModule=apache.so
LoadModule=kernel.so
LoadModule=/usr/local/lib/zabbix/dummy.so
```

Podczas uruchamiania agent załaduje moduły mariadb.so, apache.so i kernel.so z katalogu /usr/local/lib/zabbix/agent, natomiast dummy.so zostanie załadowany z /usr/local/lib/zabbix. agent nie uruchomi się, jeśli brakuje modułu, uprawnienia są nieprawidłowe lub biblioteka współdzielona nie jest modułem Zabbix.

Konfiguracja frontend

Moduły ładowalne są obsługiwane przez agent, serwer i proxy Zabbix. Dlatego typ pozycji w frontend Zabbix zależy od tego, gdzie moduł jest załadowany. Jeśli moduł jest załadowany do agenta, typ pozycji powinien być "Zabbix agent" lub "Zabbix agent (active)". Jeśli moduł jest załadowany do serwera lub proxy, typ pozycji powinien być "Simple check".

Eksport historii za pomocą modułów Zabbix nie wymaga żadnej konfiguracji frontend. Jeśli moduł zostanie pomyślnie załadowany przez serwer i udostępnia funkcję zbx_module_history_write_cbs(), która zwraca co najmniej jedną nie-NULL funkcję zwrotną, eksport historii zostanie włączony automatycznie.

Moduł dummy

Zabbix zawiera przykładowy moduł napisany w języku C. Moduł znajduje się w katalogu 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
```

Moduł jest dobrze udokumentowany i może służyć jako szablon dla własnych modułów.

Po wykonaniu ./configure w katalogu głównym drzewa źródłowego Zabbix, zgodnie z opisem powyżej, wystarczy uruchomić make, aby zbudować 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"
```

```c
/* zmienna przechowuje ustawienie limitu czasu dla przetwarzania pozycji */
static int  item_timeout = 0;

/* moduł POWINIEN definiować funkcje wewnętrzne jako static i używać wzorca nazewnictwa innego niż wewnętr
/* symbole Zabbix (zbx_*) oraz funkcje API modułów ładowanych (zbx_module_*) w celu uniknięcia konfliktów
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;
}
```

```c
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)
    {
        /* ustaw opcjonalny komunikat o błędzie */
        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)
    {
        /* ustaw opcjonalny komunikat o błędzie */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
```

```
        return SYSINFO_RET_FAIL;
    }

    param1 = get_rparam(request, 0);
    param2 = get_rparam(request, 1);

    /* dla uproszczenia nie ma ścisłej walidacji parametrów i typów */
    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)
{
    /* inicjalizacja dla 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++)
    {
        /* wykonaj coś z 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++)
    {
        /* wykonaj coś z 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++)
    {
        /* wykonaj coś z 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++)
    {
        /* wykonaj coś z 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++)
    {
        /* wykonaj coś z 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;
}
```

Moduł eksportuje trzy nowe pozycje:

- `dummy.ping` - zawsze zwraca '1'
- `dummy.echo[param1]` - zwraca pierwszy parametr bez zmian, na przykład `dummy.echo[ABC]` zwróci ABC
- `dummy.random[param1, param2]` - zwraca losową liczbę z zakresu param1-param2, na przykład `dummy.random[1,1000000]`

Ograniczenia

Obsługa modułów ładowanych jest zaimplementowana tylko dla platformy Unix. Oznacza to, że nie działa to dla agentów Windows.

W niektórych przypadkach moduł może potrzebować odczytać parametry konfiguracyjne związane z modułem z pliku `zabbix_agentd.conf`. Obecnie nie jest to obsługiwane. Jeśli potrzebujesz, aby Twój moduł używał niektórych parametrów konfiguracyjnych, powinieneś prawdopodobnie zaimplementować analizę pliku konfiguracyjnego specyficznego dla modułu.

Moduły frontend

Omówienie

Możliwe jest rozszerzenie funkcjonalności frontend Zabbix przez dodawanie modułów firm trzecich lub tworzenie własnych modułów bez konieczności zmieniania kodu źródłowego Zabbix.

Należy pamiętać, że kod modułu będzie uruchamiany z tymi samymi uprawnieniami co kod źródłowy Zabbix. Oznacza to:

- Moduły firm trzecich mogą być szkodliwe. Musisz ufać modułom, które instalujesz.
- Błędy w kodzie modułu firm trzecich mogą spowodować awarię frontend. Jeśli tak się stanie, po prostu usuń kod modułu z frontend. Po ponownym załadowaniu frontend Zabbix zobaczysz informację, że niektóre moduły są nieobecne. Przejdź do Administracja modułami (w *Administracja* > *Ogólne* > *Moduły*) i kliknij ponownie *Skanuj katalog*, aby usunąć nieistniejące moduły z bazy danych.

Instalacja

Zawsze czytaj instrukcję instalacji dla konkretnego modułu. Zaleca się instalowanie nowych modułów pojedynczo, aby łatwo wykrywać niepowodzenia.

Tuż przed instalacją modułu:

- Upewnij się, że pobrałeś moduł z zaufanego źródła. Instalacja szkodliwego kodu może prowadzić do konsekwencji, takich jak utrata danych.
- Różne wersje tego samego modułu (to samo ID) mogą być instalowane równolegle, ale tylko jedna wersja może być włączona jednocześnie.

Kroki instalacji modułu:

- Rozpakuj moduł do własnego folderu w katalogu `modules` instalacji frontend Zabbixa (na przykład `zabbix/ui/modules`).
- Upewnij się, że folder modułu zawiera co najmniej plik `manifest.json`.
- Przejdź do Administracja modułów i kliknij przycisk *Skanuj katalog*.
- Nowy moduł pojawi się na liście wraz z wersją, autorem, opisem i statusem.
- Włącz moduł, klikając jego status.

Rozwiązywanie problemów:

Problem	Rozwiązanie
Moduł nie pojawił się na liście	Jeśli moduł nie pojawił się na liście, upewnij się, że plik `manifest.json` istnieje w `modules/twój-moduł/`. Jeśli nie, mogłeś rozpakować moduł do niewłaściwego katalogu. Jeśli plik istnieje, moduł może nie być zgodny z Twoją wersją Zabbixa. Sprawdź też, czy użytkownik serwera WWW ma co najmniej uprawnienia do odczytu i przeszukiwania (`r-x`) dla folderu modułu i wszystkich podkatalogów oraz uprawnienia do odczytu (`r--`) dla wszystkich plików wewnątrz.
frontend uległ awarii	Kod modułu nie jest zgodny z bieżącą wersją Zabbixa lub konfiguracją serwera. Usuń pliki modułu i ponownie załaduj frontend. Zobaczysz powiadomienie, że niektóre moduły są niedostępne. Przejdź do Administracja modułów i ponownie kliknij *Skanuj katalog*, aby usunąć nieistniejące moduły z bazy danych.
Pojawia się komunikat o błędzie dotyczącym identycznej przestrzeni nazw, ID lub akcji	Nowy moduł próbował zarejestrować przestrzeń nazw, ID lub akcje, które są już zarejestrowane przez inne włączone moduły. Wyłącz moduł powodujący konflikt (wskazany w komunikacie o błędzie) przed włączeniem nowego.
Pojawiają się techniczne komunikaty o błędach	Zgłoś błędy twórcy modułu.

Tworzenie modułów

Aby uzyskać informacje o tworzeniu niestandardowych modułów, zobacz Developer center.

Aneksy

Przegląd Ta sekcja zawiera materiały referencyjne, parametry konfiguracyjne, specyfikacje techniczne oraz przewodniki po procedurach, które uzupełniają główny podręcznik.

Podstawowe materiały referencyjne

- Instalacja i konfiguracja — tworzenie bazy danych, instalacja i konfiguracja serwera oraz agenta.
- Konfiguracja procesów — parametry konfiguracyjne dla serwera Zabbix, proxy, agenta i innych komponentów.
- Pozycje — informacje referencyjne o kluczach pozycji i obsługiwanych typach danych.
- Makra — składnia i użycie wbudowanych makr oraz makr użytkownika.

Instalacja i konfiguracja

Ta sekcja obejmuje tworzenie bazy danych, konfigurację zabezpieczeń, procedury konfiguracji specyficzne dla platformy oraz konfigurację funkcji.

Skorzystaj z paska bocznego, aby znaleźć procedurę odpowiednią dla Twojego środowiska.

Tworzenie bazy danych

Przegląd

Baza danych Zabbix musi zostać utworzona podczas instalacji serwera Zabbix lub proxy.

Ta sekcja zawiera instrukcje tworzenia bazy danych Zabbix. Dla każdej obsługiwanej bazy danych dostępny jest osobny zestaw instrukcji.

Note:
Aby zwiększyć bezpieczeństwo bazy danych poprzez utworzenie ról/użytkowników bazy danych z minimalnymi uprawnieniami, zobacz najlepsze praktyki tworzenia bazy danych dla każdej obsługiwanej bazy danych:

- MySQL/MariaDB
- PostgreSQL/TimescaleDB

Informacje dotyczące konfigurowania bezpiecznych połączeń TLS znajdują się w sekcji Bezpieczne połączenie z bazą danych.

UTF-8 jest jedynym kodowaniem obsługiwanym przez Zabbix. Wiadomo, że działa bez żadnych luk bezpieczeństwa. Użytkownicy powinni mieć świadomość, że w przypadku używania niektórych innych kodowań występują znane problemy bezpieczeństwa. Informacje o przejściu na UTF-8 znajdują się w sekcji Naprawa zestawu znaków i sortowania bazy danych Zabbix. Zobacz także Ograniczenia filtrowania z użyciem sortowań utf8mb4.

Note:
Jeśli instalujesz z repozytorium Git Zabbix, przed przejściem do kolejnych kroków musisz uruchomić następujące polecenie:

 `make dbschema`

MySQL/MariaDB

Zestaw znaków utf8mb4 z sortowaniem utf8mb4_bin jest obsługiwany, aby serwer/proxy Zabbix działał prawidłowo z bazami danych MySQL/MariaDB.
 Aby uniknąć problemów w przyszłości, zdecydowanie zaleca się używanie zestawu znaków utf8mb4 z sortowaniem utf8mb4_bin zarówno w przypadku nowych, jak i istniejących instalacji. Aby przekonwertować bazy danych używające zestawu znaków utf8mb3 i sortowania utf8mb3_bin na zestaw znaków utf8mb4 i sortowanie utf8mb4_bin, postępuj zgodnie z instrukcjami w sekcji Naprawianie zestawu znaków i sortowania bazy danych Zabbix.

Wyzwalacze deterministyczne muszą zostać utworzone podczas importowania schematu. W MySQL i MariaDB wymaga to ustawienia `GLOBAL log_bin_trust_function_creators = 1`, jeśli rejestrowanie binarne jest włączone, nie ma uprawnień superużytkownika, a `log_bin_trust_function_creators = 1` nie jest ustawione w pliku konfiguracyjnym MySQL.

Jeśli instalujesz Zabbix z pakietów lub kontenerów, zapoznaj się z odpowiednimi instrukcjami.

Jeśli instalujesz Zabbix ze źródeł:

- Utwórz i skonfiguruj bazę danych oraz użytkownika. Ze względów bezpieczeństwa użyj silnego, unikatowego hasła dla użytkownika bazy danych 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;
```

- Zaimportuj dane do bazy danych i ustaw zestaw znaków utf8mb4 jako domyślny (zakładając, że znajdujesz się w katalogu głównym źródeł Zabbix). W przypadku bazy danych proxy Zabbix należy zaimportować tylko `schema.sql` (bez images.sql i data.sql).

```
cd database/mysql
mysql -uzabbix -p<password> zabbix < schema.sql
#### zatrzymaj się tutaj, jeśli tworzysz bazę danych dla proxy Zabbix
mysql -uzabbix -p<password> zabbix < images.sql
mysql -uzabbix -p<password> --default-character-set=utf8mb4 zabbix < data.sql
```

`log_bin_trust_function_creators` można wyłączyć po pomyślnym zaimportowaniu schematu:

```
mysql -uroot -p<password>

mysql> SET GLOBAL log_bin_trust_function_creators = 0;
mysql> quit;
```

PostgreSQL

Musisz mieć użytkownika bazy danych z uprawnieniami do tworzenia obiektów bazy danych.

Jeśli instalujesz Zabbix z pakietów lub kontenerów, zapoznaj się z odpowiednimi instrukcjami.

Jeśli instalujesz Zabbix ze źródeł:

- Utwórz użytkownika bazy danych.

Poniższe polecenie powłoki utworzy użytkownika `zabbix`. Podaj hasło, gdy zostaniesz o nie poproszony, i powtórz hasło (uwaga: najpierw możesz zostać poproszony o hasło `sudo`):

```
sudo -u postgres createuser --pwprompt zabbix
```

- Utwórz bazę danych.

Poniższe polecenie powłoki utworzy bazę danych `zabbix` (ostatni parametr) z wcześniej utworzonym użytkownikiem jako właścicielem (`-O zabbix`).

```
sudo -u postgres createdb -O zabbix -E Unicode -T template0 zabbix
```

- Zaimportuj początkowy schemat i dane (zakładając, że znajdujesz się w katalogu głównym źródeł Zabbix). W przypadku bazy danych Zabbix proxy należy zaimportować tylko `schema.sql` (bez `images.sql` ani `data.sql`).

```
cd database/postgresql
cat schema.sql | sudo -u zabbix psql zabbix
#### zatrzymaj się tutaj, jeśli tworzysz bazę danych dla Zabbix proxy
cat images.sql | sudo -u zabbix psql zabbix
cat data.sql | sudo -u zabbix psql zabbix
```

Attention:
Powyższe polecenia są podane jako przykład, który zadziała w większości instalacji GNU/Linux. Możesz użyć innych poleceń w zależności od konfiguracji systemu/bazy danych, na przykład:

 `psql -U <username>`

 Jeśli masz jakiekolwiek problemy z konfiguracją bazy danych, skonsultuj się z administratorem bazy danych.

TimescaleDB

Instrukcje dotyczące tworzenia i konfiguracji TimescaleDB znajdują się w osobnej sekcji.

Elasticsearch i ClickHouse

Zabbix może również przechowywać dane historyczne w Elasticsearch lub ClickHouse. Instrukcje konfiguracji znajdują się na stronach podanych w linkach.

SQLite

Używanie SQLite jest obsługiwane tylko dla **Zabbix proxy**!

Baza danych zostanie utworzona automatycznie, jeśli nie istnieje.

Wróć do sekcji instalacji.

Naprawa zestawu znaków i sortowania bazy danych Zabbix

MySQL/MariaDB

Historycznie MySQL i jego pochodne używały `utf8` jako aliasu dla utf8mb3 - własnej 3-bajtowej implementacji standardowego UTF8, które jest 4-bajtowe. Począwszy od MySQL 8.0.28 i MariaDB 10.6.1, zestaw znaków `utf8mb3` jest przestarzały i w pewnym momencie jego obsługa zostanie usunięta, a `utf8` stanie się odwołaniem do `utf8mb4`. Od Zabbix 6.0 obsługiwane jest `utf8mb4`. Aby uniknąć przyszłych problemów, zdecydowanie zaleca się używanie `utf8mb4`. Dodatkową zaletą przejścia na `utf8mb4` jest obsługa dodatkowych znaków Unicode.

Warning:
Ponieważ wersje wcześniejsze niż Zabbix 6.0 nie obsługują `utf8mb4`, przed wykonaniem konwersji `utf8mb4` upewnij się, że najpierw zaktualizowano serwer Zabbix oraz schemat bazy danych do wersji 6.0.x lub nowszej.

1. Sprawdź zestaw znaków i sortowanie bazy danych.

Na przykład:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
```

```
| latin2                   | latin2 _general_ci   |
+--------------------------+----------------------+
```

Lub:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8                     | utf8_bin             |
+--------------------------+----------------------+
```

Jak widać, zestaw znaków nie jest tutaj ustawiony na `utf8mb4`, a sortowanie nie jest `utf8mb4\_bin`, więc należy to poprawić.

2. Zatrzymaj Zabbix.

3. Utwórz kopię zapasową bazy danych!

4. Popraw zestaw znaków i sortowanie na poziomie bazy danych:

```
alter database <your DB name> character set utf8mb4 collate utf8mb4_bin;
```

Poprawione wartości:

```
mysql> SELECT @@character_set_database, @@collation_database;
+--------------------------+----------------------+
| @@character_set_database | @@collation_database |
+--------------------------+----------------------+
| utf8mb4                  | utf8mb4_bin          |
+--------------------------+----------------------+
```

5. Załaduj skrypt , aby poprawić zestaw znaków i sortowanie na poziomie tabel i kolumn:

```
mysql <your DB name> < utf8mb4_convert.sql
```

6. Wykonaj skrypt:

```
                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;
```

Należy pamiętać, że `utf8mb4` powinno zajmować nieco więcej miejsca na dysku.

7. Jeśli nie wystąpiły błędy - możesz utworzyć kopię zapasową bazy danych z poprawioną konfiguracją.

8. Uruchom Zabbix.

Aktualizacja bazy danych do kluczy głównych

Przegląd

Ta sekcja zawiera instrukcje ręcznej aktualizacji tabel w istniejących instalacjach do kluczy głównych.

Aktualizacja do kluczy głównych optymalizuje sposób indeksowania i dostępu do danych, co może przyspieszyć zapytania i oszczędzić miejsce. Poprawia także zarządzanie danymi i synchronizację w konfiguracjach klastrowych, pomagając w skalowaniu i zapewniając niezawodność systemu nawet w przypadku awarii niektórych serwerów.

> **Attention:**
> Instrukcje podane na tej stronie są przeznaczone dla zaawansowanych użytkowników i mogą wymagać dostosowania do konkretnej konfiguracji. Aktualizacja do kluczy głównych może być czasochłonna i zasobożerna. Upewnij się, że dostępna jest wystarczająca ilość wolnego miejsca na dysku; w zależności od rozmiaru bazy danych i przechowywanych danych proces może wymagać nawet do 2,5 razy więcej miejsca niż obecnie zajmują tabele historii.

Klucze główne są używane we wszystkich tabelach w nowych instalacjach od Zabbix 6.0.

Nie ma automatycznej aktualizacji bazy danych do kluczy głównych; jednak istniejące instalacje można zaktualizować ręcznie **po** uaktualnieniu serwera Zabbix do wersji 6.0 lub nowszej.

Attention:
Od Zabbix 7.0 aktualizacja tabel do kluczy głównych obejmuje również zmianę tabel na typy danych double precision.

 Jeśli używasz Zabbix 7.0 (lub nowszego), tabele już korzystają z double precision. Jednak instrukcje na tej stronie nadal można zastosować, aby zaktualizować tabele do kluczy głównych bez wpływu na tabele, które już używają double precision.

 Jeśli używasz Zabbix 6.4 (lub starszego), rozważ najpierw aktualizację tabel do double precision. Więcej informacji znajdziesz w sekcji Aktualizacja do wartości numerycznych o rozszerzonym zakresie w dokumentacji Zabbix 7.0.

Dostępne są instrukcje dla:

- MariaDB/MySQL
- PostgreSQL
- TimescaleDB

Ważne uwagi

Aby przeprowadzić aktualizację bazy danych:

1. Zatrzymaj serwer Zabbix.

Zdecydowanie zaleca się zatrzymanie serwera Zabbix na czas aktualizacji. Jeśli jednak jest to absolutnie konieczne, możesz przeprowadzić aktualizację, gdy serwer działa (dotyczy tylko MySQL, MariaDB oraz PostgreSQL bez TimescaleDB).

2. Wykonaj kopię zapasową bazy danych.
3. Zainstaluj najnowszy pakiet zabbix-sql-scripts zgodny z używaną wersją Zabbix (np. dla RHEL: `dnf install zabbix-sql-scripts`).
4. Uruchom skrypty dla swojej bazy danych.
5. Uruchom serwer Zabbix.

Warning:
Uruchom skrypty tylko dla bazy danych serwera. proxy nie odniesie korzyści z tej aktualizacji.

Jeśli baza danych używa partycji, skontaktuj się z administratorem DB lub pomocą techniczną Zabbix.

Pliki CSV można usunąć po pomyślnej aktualizacji do kluczy głównych.

Opcjonalnie frontend Zabbix można przełączyć w tryb konserwacji.

MariaDB/MySQL

Eksport i import należy wykonywać w tmux/screen, aby upewnić się, że sesja nie zostanie przerwana.

Zobacz także: Important notes

MySQL 8.4+ z mysqlsh

Ta metoda może być używana przy działającym serwerze Zabbix, jednak zaleca się zatrzymanie serwera na czas aktualizacji. MySQL Shell (*mysqlsh*) musi być zainstalowany i musi mieć możliwość połączenia z bazą danych.

- Zaloguj się do konsoli MySQL jako root (zalecane) lub jako dowolny użytkownik z uprawnieniami FILE.
- Uruchom MySQL z włączoną zmienną local_infile.
- Zmień nazwy starych tabel i utwórz nowe tabele, uruchamiając `history_upgrade_prepare.sql`.

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

- Wyeksportuj i zaimportuj dane.

Połącz się przez mysqlsh. Jeśli używasz połączenia przez gniazdo, może być wymagane podanie ścieżki.

```
sudo mysqlsh -uroot -S /run/mysqld/mysqld.sock --no-password -Dzabbix
```

Przełącz się do trybu JavaScript za pomocą:

```
\js
```

Następnie uruchom poniższy kod (wartość CSVPATH można w razie potrzeby zmienić):

```
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" });
```

Jeśli pojawi się komunikat "JavaScript is not supported", oznacza to, że instalacja MySQL Shell nie obsługuje JS. W takim przypadku zainstaluj oficjalny pakiet MySQL Shell firmy Oracle (lub zbuduj go ze źródeł), aby włączyć tryb JavaScript.

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

MariaDB/MySQL 8.4+ bez mysqlsh

Ta metoda aktualizacji zajmuje więcej czasu i powinna być używana tylko wtedy, gdy aktualizacja z użyciem *mysqlsh* nie jest możliwa.

Aktualizacja tabel

- Zaloguj się do konsoli MySQL jako root (zalecane) lub dowolny użytkownik z uprawnieniami FILE.
- Jeśli wykonujesz migrację przy działającym serwerze Zabbix, uruchom MySQL z włączoną zmienną local_infile.
- Zmień nazwy starych tabel i utwórz nowe tabele, uruchamiając `history_upgrade_prepare.sql`:

```
mysql -uzabbix -p<password> zabbix < /usr/share/zabbix/sql-scripts/mysql/option-patches/history_upgrade_pr
```

Migracja przy zatrzymanym serwerze

`max_execution_time` (w MySQL) lub `max_statement_time` (w MariaDB) musi zostać wyłączony przed migracją danych, aby uniknąć przekroczenia limitu czasu podczas migracji.

Dla MySQL:

```
SET @@max_execution_time=0;
```

Dla 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;
```

Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

Migracja przy działającym serwerze

Sprawdź, dla których ścieżek włączony jest import/eksport:

```
mysql> SELECT @@secure_file_priv;
+-----------------------+
| @@secure_file_priv    |
+-----------------------+
| /var/lib/mysql-files/ |
+-----------------------+
```

Jeśli wartość *secure_file_priv* jest ścieżką do katalogu, eksport/import będzie wykonywany dla plików w tym katalogu. W takim przypadku odpowiednio zmień ścieżki do plików w zapytaniach albo ustaw wartość *secure_file_priv* na pusty ciąg na czas aktualizacji.

Jeśli wartość *secure_file_priv* jest pusta, eksport/import może być wykonywany z dowolnej lokalizacji.

Jeśli wartość *secure_file_priv* ma wartość NULL, ustaw ją na ścieżkę zawierającą wyeksportowane dane tabel ('/var/lib/mysql-files/' w powyższym przykładzie).

Więcej informacji znajdziesz w dokumentacji MySQL lub dokumentacji MariaDB.

`max_execution_time` (w MySQL) lub `max_statement_time` (w MariaDB) musi zostać wyłączony przed eksportem danych, aby uniknąć przekroczenia limitu czasu podczas eksportu.

Dla MySQL:

```
SET @@max_execution_time=0;
```

Dla 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
```

Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

PostgreSQL

Eksport i import należy wykonać w tmux/screen, aby upewnić się, że sesja nie zostanie przerwana. W przypadku instalacji z TimescaleDB pomiń tę sekcję i przejdź do PostgreSQL + TimescaleDB.

Zobacz także: Ważne uwagi

Aktualizacja tabel

- Zmień nazwy tabel, używając `history_upgrade_prepare.sql`:

```
sudo -u zabbix psql zabbix < /usr/share/zabbix/sql-scripts/postgresql/option-patches/history_upgrade_prepa
```

Migracja przy zatrzymanym serwerze

- Wyeksportuj bieżącą historię, zaimportuj ją do tabeli tymczasowej, a następnie wstaw dane do nowych tabel, ignorując duplikaty:

```
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;
```

Zobacz wskazówki dotyczące poprawy wydajności INSERT: PostgreSQL: Bulk Loading Huge Amounts of Data, Checkpoint Distance and Amount of WAL.

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

Migracja z działającym serwerem

- Wyeksportuj bieżącą historię, zaimportuj ją do tabeli tymczasowej, a następnie wstaw dane do nowych tabel, ignorując duplikaty:

```
\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;
```

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

PostgreSQL + TimescaleDB

Eksport i import muszą być wykonywane w tmux/screen, aby upewnić się, że sesja nie zostanie przerwana. Serwer Zabbix powinien być wyłączony podczas aktualizacji.

Zobacz także: Important notes

- Zmień nazwy tabel, używając `history_upgrade_prepare.sql`.
  - Jeśli kompresja jest włączona (w domyślnej instalacji), uruchom skrypt z `/usr/share/zabbix/sql-scripts/postgresql/tim`

    ```
    cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/with-compression/history_
    ```

  - Jeśli kompresja jest wyłączona, uruchom skrypt z `/usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-p`

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-patches/without-compression/histo
```

- Uruchom skrypty migracji hypertable TimescaleDB w zależności od ustawień kompresji:
  - Jeśli kompresja jest włączona (w domyślnej instalacji), uruchom skrypty z `/usr/share/zabbix/sql-scripts/postgresql/ti`

```
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
```

  - Jeśli kompresja jest wyłączona, uruchom skrypty z `/usr/share/zabbix/sql-scripts/postgresql/timescaledb/option-`

```
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
```

Zobacz także: Tips dotyczące poprawy wydajności INSERT.

- Postępuj zgodnie z instrukcjami po migracji, aby usunąć stare tabele.

Po migracji

Dla wszystkich baz danych, po zakończeniu migracji wykonaj następujące czynności:

- Zweryfikuj, że wszystko działa zgodnie z oczekiwaniami.
- Usuń stare tabele:

```
DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;
```

- W przypadku TimescaleDB usuń również następującą starą tabelę:

```
DROP TABLE trends_old;
```

Zobacz także

- Przygotowanie tabeli auditlog do partycjonowania

Przygotowanie tabeli auditlog do partycjonowania

Przegląd

Niektóre bazy danych (na przykład MySQL) wymagają, aby kolumna partycjonowania była częścią unikalnego ograniczenia tabeli. Dlatego, aby partycjonować tabelę `auditlog` według czasu, klucz główny musi zostać zmieniony z `auditid` na klucz złożony `auditid` + `clock`.

Ta sekcja zawiera instrukcje dotyczące zmiany klucza głównego tabeli `auditlog`.

Attention:
Instrukcje podane na tej stronie są przeznaczone dla zaawansowanych użytkowników. Należy pamiętać, że instrukcje te mogą wymagać dostosowania do konkretnej konfiguracji. Zmiana klucza głównego może być również niezgodna z przyszłymi poprawkami aktualizacyjnymi, dlatego może być konieczna ręczna obsługa przyszłych aktualizacji.

 Zmiana klucza głównego może być operacją wymagającą dużych zasobów i zająć dużo czasu, w zależności od rozmiaru tabeli `auditlog`. Zaleca się zatrzymanie serwera Zabbix i przełączenie frontend Zabbix w tryb konserwacji na czas wprowadzania zmiany. Jeśli jednak jest to absolutnie konieczne, istnieje sposób na zmianę klucza głównego bez przestoju (patrz poniżej).

Partycjonowanie tabeli `auditlog` może poprawić na przykład housekeeping w dużych środowiskach. Chociaż housekeeping Zabbix obecnie nie może korzystać z tabel partycjonowanych (z wyjątkiem TimescaleDB), można wyłączyć housekeeping Zabbix i usuwać partycje za pomocą skryptów.

Od Zabbix 7.0 tabela auditlog dla TimescaleDB została przekształcona w hypertable, co pozwala housekeeperowi usuwać dane całymi chunkami. Aby zaktualizować istniejącą tabelę auditlog do hypertable, zobacz Upgrading TimescaleDB schema.

MySQL

Ważne uwagi dotyczące przebudowy indeksów

MySQL automatycznie przebudowuje indeksy dla klucza głównego podczas operacji ALTER TABLE. Jednak zdecydowanie zaleca się również ręczną przebudowę indeksów za pomocą polecenia OPTIMIZE TABLE, aby zapewnić optymalną wydajność bazy danych.

Przebudowa indeksów może tymczasowo wymagać dodatkowej ilości miejsca na dysku równej rozmiarowi samej tabeli. Aby uzyskać bieżący rozmiar danych i indeksów, można wykonać następujące polecenia:

```
ANALYZE TABLE auditlog;
SHOW TABLE STATUS LIKE 'auditlog';
```

Jeśli dostępna przestrzeń dyskowa jest problemem, postępuj zgodnie z instrukcjami Zmiana klucza głównego bez przestoju. Dostępne są również inne opcje:

- Zwiększenie parametru MySQL sort_buffer_size może pomóc zmniejszyć zużycie miejsca na dysku podczas ręcznej przebudowy indeksów. Należy jednak pamiętać, że modyfikacja tej zmiennej może wpłynąć na ogólne zużycie pamięci przez bazę danych.
- Rozważ zwolnienie miejsca poprzez usunięcie potencjalnie niepotrzebnych danych.
- Rozważ zmniejszenie parametru *Okres przechowywania danych* housekeeper przed uruchomieniem housekeepera.

Zmiana klucza głównego z przestojem

1. Usuń bieżący klucz główny tabeli auditlog i dodaj nowy klucz główny.

```
ALTER TABLE auditlog DROP PRIMARY KEY, ADD PRIMARY KEY (auditid, clock);
```

2. Odbuduj indeksy (opcjonalnie, ale zdecydowanie zalecane, zobacz Ważne uwagi dotyczące odbudowy indeksów).

```
OPTIMIZE TABLE auditlog;
```

Zmiana klucza głównego bez przestoju

Ręczna metoda zmiany klucza głównego jest opisana tutaj. Alternatywnie można użyć zestawu narzędzi pt-online-schema-change firmy Percona. Ten zestaw narzędzi automatycznie wykonuje następujące działania, jednocześnie minimalizując ilość miejsca używanego podczas zmiany tabeli auditlog.

1. Utwórz nową tabelę z nowym kluczem głównym i utwórz indeksy.

```
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. Zamień tabele.

```
RENAME TABLE auditlog TO auditlog_old, auditlog_new TO auditlog;
```

3. Skopiuj dane ze starej tabeli do nowej tabeli.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Można to wykonać w częściach (za pomocą wielu instrukcji INSERT INTO z klauzulami WHERE clock, w razie potrzeby), aby uniknąć nadmiernego zużycia zasobów.

4. Usuń starą tabelę.

```
DROP TABLE auditlog_old;
```

PostgreSQL

Ważne uwagi dotyczące przebudowy indeksów

PostgreSQL automatycznie przebudowuje indeksy dla klucza głównego podczas operacji `ALTER TABLE`. Jednak zdecydowanie zaleca się również ręczną przebudowę indeksów za pomocą polecenia `REINDEX TABLE CONCURRENTLY`, aby zapewnić optymalną wydajność bazy danych.

Przebudowa indeksów może tymczasowo wymagać nawet trzykrotności miejsca na dysku aktualnie zajmowanego przez indeksy. Aby sprawdzić bieżący rozmiar indeksów, możesz wykonać następujące zapytanie:

```
SELECT pg_size_pretty(pg_indexes_size('auditlog'));
```

Jeśli dostępne miejsce na dysku jest problemem, postępuj zgodnie z instrukcjami w sekcji Zmiana klucza głównego bez przestoju. Dostępne są również inne opcje:

- Zwiększenie parametru PostgreSQL `maintenance_work_mem` może pomóc zmniejszyć zużycie miejsca na dysku podczas ręcznej przebudowy indeksów. Jednak modyfikacja tej zmiennej może wpłynąć na ogólne zużycie pamięci bazy danych.
- Jeśli masz inny dysk lub tablespace z większą ilością wolnego miejsca, możesz rozważyć zmianę lokalizacji tymczasowego przechowywania na potrzeby przebudowy indeksu. Możesz ustawić parametr PostgreSQL `temp_tablespaces`, aby wskazać inny tablespace dla obiektów tymczasowych.
- Rozważ zwolnienie miejsca poprzez usunięcie potencjalnie niepotrzebnych danych.
- Rozważ zmniejszenie parametru *Okres przechowywania danych* housekeeper przed uruchomieniem housekeeper.

Zmiana klucza głównego z przestojem

1. Usuń bieżący klucz główny tabeli `auditlog` i dodaj nowy klucz główny.

```
ALTER TABLE auditlog DROP CONSTRAINT auditlog_pkey;
ALTER TABLE auditlog ADD PRIMARY KEY (auditid,clock);
```

2. Odbuduj indeksy (opcjonalnie, ale zdecydowanie zalecane; zobacz Ważne uwagi dotyczące odbudowy indeksów).

```
REINDEX TABLE CONCURRENTLY auditlog;
```

Zmiana klucza głównego bez przestoju

Opis ręcznej metody zmiany klucza głównego znajduje się tutaj. Alternatywnie można rozważyć rozszerzenie `pg_repack` do utworzenia nowej tabeli, skopiowania danych i zamiany tabel.

1. Utwórz nową tabelę z nowym kluczem głównym i utwórz indeksy.

```
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. Zamień tabele.

```
ALTER TABLE auditlog RENAME TO auditlog_old;
ALTER TABLE auditlog_new RENAME TO auditlog;
```

3. Skopiuj dane ze starej tabeli do nowej tabeli.

```
INSERT INTO auditlog SELECT * FROM auditlog_old;
```

Można to wykonać w częściach (wiele instrukcji `INSERT INTO` z klauzulami `WHERE clock` w razie potrzeby), aby uniknąć nadmiernego zużycia zasobów.

4. Usuń starą tabelę.

```
DROP TABLE auditlog_old;
```

Zobacz także

- Aktualizacja bazy danych do kluczy głównych

Konfiguracja TimescaleDB

Przegląd

Zabbix obsługuje TimescaleDB, rozwiązanie bazodanowe oparte na PostgreSQL, które automatycznie partycjonuje dane na fragmenty oparte na czasie, aby zapewnić wyższą wydajność przy dużej skali.

Warning:
Obecnie TimescaleDB nie jest obsługiwany przez Zabbix proxy.

Instrukcje na tej stronie można wykorzystać w następujących scenariuszach:

- Tworzenie bazy danych TimescaleDB lub migracja z istniejących tabel PostgreSQL do TimescaleDB (zobacz Konfiguracja).
- Aktualizacja schematu istniejącej bazy danych TimescaleDB podczas aktualizacji Zabbix (zobacz Aktualizacja schematu TimescaleDB).

Konfiguracja

Wymagania wstępne: Rozszerzenie TimescaleDB w obsługiwanej wersji zainstalowane na serwerze bazy danych. Instrukcje instalacji znajdują się w dokumentacji TimescaleDB.

Warning:
Przed instalacją TimescaleDB zainstaluj obsługiwaną wersję PostgreSQL z oficjalnego repozytorium PostgreSQL.

Włącz rozszerzenie TimescaleDB dla określonej bazy danych, wykonując:

```
echo "CREATE EXTENSION IF NOT EXISTS timescaledb CASCADE;" | sudo -u postgres psql zabbix
```

Uruchomienie tego polecenia wymaga uprawnień administratora bazy danych.

Note:
Jeśli używasz schematu bazy danych innego niż 'public', musisz dodać klauzulę SCHEMA do powyższego polecenia. Np.:
 `echo "CREATE EXTENSION IF NOT EXISTS timescaledb SCHEMA yourschema CASCADE;" | sudo -u postgres psql zabbix`

Następnie uruchom skrypt `postgresql/timescaledb/schema.sql`. W przypadku nowych instalacji skrypt należy uruchomić po utworzeniu standardowej bazy danych PostgreSQL z początkowym schematem/danymi (zobacz tworzenie bazy danych).

```
cat /usr/share/zabbix/sql-scripts/postgresql/timescaledb/schema.sql | sudo -u zabbix psql zabbix
```

Attention:
Zignoruj komunikaty ostrzegawcze informujące o nieprzestrzeganiu najlepszych praktyk podczas uruchamiania skryptu `schema.sql` w TimescaleDB w wersji 2.9.0 i nowszej. Niezależnie od tego ostrzeżenia konfiguracja zostanie ukończona pomyślnie.

Migracja istniejących danych historii, trendów i dziennika audytu może zająć dużo czasu. serwer Zabbix i frontend muszą być wyłączone na czas migracji.

Skrypt `schema.sql` ustawia następujące parametry housekeeping:

- Zastąp okres przechowywania historii pozycji
- Zastąp okres przechowywania trendów pozycji

Aby używać partycjonowanego housekeeping dla historii i trendów, obie te opcje muszą być włączone. Możliwe jest również włączenie nadpisania indywidualnie tylko dla historii albo tylko dla trendów.

Dla PostgreSQL i TimescaleDB skrypt `postgresql/timescaledb/schema.sql` ustawia dwa dodatkowe parametry:

- Włącz kompresję
- Kompresuj rekordy starsze niż 7 dni

Aby housekeeper mógł pomyślnie usuwać skompresowane dane, muszą być włączone obie opcje: *Override item history period* oraz *Override item trend period*. Jeśli nadpisywanie jest wyłączone, a tabele zawierają skompresowane fragmenty, housekeeper nie będzie usuwał danych z tych tabel, a ostrzeżenia o nieprawidłowej konfiguracji będą wyświetlane w sekcjach *Housekeeping* oraz *System information*.

Wszystkie te parametry można zmienić w *Administracja* > *Housekeeping* po instalacji.

Note:
Możesz uruchomić narzędzie timescaledb-tune dostarczane przez TimescaleDB, aby zoptymalizować parametry konfiguracji PostgreSQL w pliku `postgresql.conf`.

Aktualizacja schematu TimescaleDB

Podczas aktualizacji Zabbix do wersji zawierającej nowe hypertabele TimescaleDB, serwer Zabbix nie konfiguruje automatycznie tych hypertabel (na przykład podczas aktualizacji z Zabbix 6.4 do 8.0, ponieważ wersje 7.0.0 i 7.0.2 wprowadziły nowe hypertabele).

Aby skonfigurować nowe hypertabele TimescaleDB, wykonaj następujące kroki:

1. Uruchom serwer Zabbix; spowoduje to aktualizację istniejącej bazy danych.
2. Sprawdź w pliku dziennika serwera, czy aktualizacja bazy danych została zakończona; po jej zakończeniu zatrzymaj serwer Zabbix. Zwróć uwagę, że serwer zapisuje ostrzeżenie, jeśli próbuje włączyć kompresję dla tabeli, która nie jest hypertabelą.
3. Uruchom skrypt `postgresql/timescaledb/schema.sql`; spowoduje to skonfigurowanie nowych hypertabel TimescaleDB. Zwróć uwagę, że od Zabbix 7.0.0 lokalizacja i nazwa skryptu zostały zmienione z `postgresql/timescaledb.sql` na `postgresql/timescaledb/schema.sql`.

Attention:
Zignoruj komunikaty ostrzegawcze informujące o nieprzestrzeganiu najlepszych praktyk podczas uruchamiania skryptu `schema.sql` w TimescaleDB w wersji 2.9.0 i nowszej. Niezależnie od tego ostrzeżenia konfiguracja zostanie pomyślnie zakończona.

Kompresja TimescaleDB

Natywna kompresja TimescaleDB jest obsługiwana dla wszystkich tabel Zabbix, które są hypertabelami TimescaleDB. Podczas aktualizacji lub migracji do TimescaleDB początkowa kompresja dużych tabel może zająć dużo czasu.

Należy pamiętać, że kompresja jest obsługiwana w ramach licencji społecznościowej Timescale ”timescale” i nie jest obsługiwana w ramach licencji Apache 2.0 ”apache”. Jeśli Zabbix wykryje, że kompresja nie jest obsługiwana, do logu serwera Zabbix zostanie zapisany komunikat ostrzegawczy, a użytkownicy nie będą mogli włączyć kompresji w frontend.

Note:
Zaleca się zapoznanie z kompresją w dokumentacji TimescaleDB przed jej użyciem.

Należy pamiętać, że kompresja nakłada pewne ograniczenia, a w szczególności:

- Modyfikacje skompresowanych chunków (wstawianie, usuwanie, aktualizowanie) nie są dozwolone
- Zmiany schematu dla skompresowanych tabel nie są dozwolone.

Ustawienia kompresji można zmienić w bloku *Kompresja historii, trendów i dziennika audytu* w sekcji *Administracja* > *Housekeeping* frontend Zabbix.

Parameter	Default	Comments
Enable compression	Enabled	Zaznaczenie lub odznaczenie pola wyboru nie aktywuje/dezaktywuje kompresji natychmiast. Ponieważ kompresją zarządza Housekeeper, zmiany zaczną obowiązywać w ciągu maksymalnie 2 razy `HousekeepingFrequency` godzin (ustawione w `zabbix_server.conf`) Po wyłączeniu kompresji nowe chunki, które trafią do okresu kompresji, nie będą kompresowane. Jednak wszystkie wcześniej skompresowane dane pozostaną skompresowane. Aby zdekompresować wcześniej skompresowane chunki, postępuj zgodnie z instrukcjami w dokumentacji TimescaleDB. Podczas aktualizacji ze starszych wersji Zabbix z obsługą TimescaleDB kompresja nie będzie domyślnie włączona.
Compress records older than	7d	Ten parametr nie może być mniejszy niż 7 dni. Ze względu na niezmienność skompresowanych chunków wszystkie późne dane (np. dane opóźnione przez proxy), które są starsze niż ta wartość, zostaną odrzucone.

Aby poprawić wydajność aktualizacji trendów, można obniżyć wartość "chunk_time_interval" dla tabel `trends` i `trends_uint` z 30 dni do 7 dni lub mniej, w zależności od liczby pozycji korzystających z trendów. Celem tego ustawienia jest dostosowanie się do najlepszych praktyk TimescaleDB oraz zapewnienie, że rozmiar chunków pozostaje w granicach zasobów dostępnych w systemie.

Konfiguracja Elasticsearch

Zabbix może przechowywać dane historyczne w Elasticsearch jako alternatywę dla relacyjnej bazy danych.

Attention:
Obsługa Elasticsearch jest obecnie eksperymentalna.

Ten przewodnik obejmuje konfigurację obsługiwanych wersji Elasticsearch. Jeśli używasz innej wersji, niektóre funkcje mogą nie działać zgodnie z oczekiwaniami.

Elasticsearch może przechowywać następujące typy wartości:

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 nie akceptuje tablic JSON. Wartość JSON musi być pojedynczym obiektem albo zbiorem obiektów.

Ważne uwagi

- Elasticsearch wymaga libcurl. Szczegóły znajdują się w sekcji wymagania.
- housekeeper nie usuwa danych z Elasticsearch. Aby kontrolować, jak długo dane są przechowywane, zobacz Index Lifecycle Management.
- Zabbix **nie** oblicza ani nie przechowuje trendów w Elasticsearch. Rozważ wydłużenie okresu przechowywania historii, aby zachować starsze dane.
- Gdy używany jest Elasticsearch, zapytania zakresowe pobierające wartości z bazy danych są ograniczone przez znacznik czasu okresu przechowywania danych.
- Elasticsearch nie jest obsługiwany dla proxy Zabbix.

Jeśli Elasticsearch nie jest jeszcze zainstalowany, przed kontynuowaniem zapoznaj się z oficjalnym przewodnikiem instalacji.

Konfigurowanie Elasticsearch

Aby przechowywać dane historii w Elasticsearch, należy:

- Utworzyć indeks dla każdego typu wartości, który chcesz przechowywać — to tutaj Elasticsearch zapisuje dane, podobnie jak tabela w relacyjnej bazie danych.
- Zdefiniować mapowanie dla każdego indeksu — określa ono strukturę danych, podobnie jak schemat tabeli.
- Skonfigurować pipeline przetwarzania, aby przetwarzać wartości przed zapisaniem (wymagane dla wartości JSON i indeksów opartych na dacie).

Elasticsearch może przechowywać dane w jednym indeksie dla każdego typu wartości lub w wielu indeksach opartych na dacie. Oba podejścia opisano poniżej.

Przechowywanie historii w pojedynczym indeksie

W tym podejściu wszystkie dane historii dla danego typu wartości są zapisywane w jednym indeksie (np. `uint` lub `text`).

Aby utworzyć indeks dla typu wartości *Numeric (unsigned)*, wyślij następujące żądanie (z `/uint` w adresie URL) do swojej instancji 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 odpowie potwierdzeniem, że indeks został utworzony:

```
{"acknowledged": true, "shards_acknowledged": true, "index": "uint"}
```

Podobne żądania należy wysłać dla każdego dodatkowego typu wartości, który chcesz przechowywać w Elasticsearch.

> **Note:**
> Mapowania dla wszystkich typów wartości są dostępne w repozytorium źródłowym Zabbix.

Na przykład, aby utworzyć indeks dla typu wartości *Text*:

```
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"
         }
      }
   }
}'
```

Typ wartości JSON

W przeciwieństwie do innych typów wartości, wartości JSON wymagają dodatkowego przetwarzania przed zapisaniem.

Poniższy indeks używa oddzielnych pól dla sparsowanych i surowych wartości, dlatego do sparsowania każdej wartości jako JSON i zapisania jej w odpowiednim polu potrzebny jest ingest pipeline.

Aby utworzyć indeks dla typu wartości *JSON*, wyślij następujące żądanie (z `/json` w adresie URL) do swojej instancji 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 }
    }
  }
}'
```

Następnie utwórz 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 odpowie potwierdzeniem, że ingest pipeline został utworzony:

```
{"acknowledged": true}
```

Przechowywanie historii w indeksach opartych na dacie

Zamiast zapisywać wszystkie dane historyczne do jednego indeksu (np. `uint`), Elasticsearch może rozdzielać te dane między wiele indeksów opartych na dacie (np. `uint-2026-01-01`, `uint-2026-01-02`). Ułatwia to zarządzanie wolumenem danych oraz ich retencją w czasie.

Aby to włączyć, należy:

- Utworzyć szablon indeksu dla każdego typu wartości, który chcesz przechowywać — informuje to Elasticsearch, jakie mapowanie zastosować podczas automatycznego tworzenia nowego indeksu opartego na dacie.
- Utworzyć pipeline ingest dla każdego typu wartości — przetwarza on każdą przychodzącą wartość i kieruje ją do właściwego indeksu opartego na dacie.

Szablony indeksów

Aby utworzyć szablon dla indeksu `text`, wyślij żądanie z następującymi szczegółami:

- Użyj `_template/text_template` w adresie URL swojej instancji Elasticsearch.
- Użyj `"text*"` w polu `"index_patterns"`, aby dopasować nazwę indeksu.
- Użyj mapowania dla typu wartości `text` (zobacz mapowania w repozytorium źródłowym 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"
         }
      }
   }
}'
```

Szablon dla indeksu `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 }
      }
   }
}'
```

Potoki ingest

Aby utworzyć potok ingest dla indeksu `text`:

- Użyj `_ingest/pipeline/text-pipeline` w adresie URL swojej instancji Elasticsearch.
- Dołącz procesor `date_index_name`, aby kierować każdą wartość do właściwego indeksu opartego na dacie na podstawie jej znacznika czasu.

```
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"
         }
      }
   ]
}'
```

W przypadku indeksu `json` potok musi również przeanalizować wartość JSON przed skierowaniem jej do właściwego indeksu:

```
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"
         }
      }
   ]
}'
```

Konfigurowanie serwera Zabbix

W pliku konfiguracyjnym serwera Zabbix (`zabbix_server.conf`) ustaw parametr `HistoryProvider`.

Na przykład, aby przechowywać wartości typu *Character*, *Log*, *Text* i *JSON* w Elasticsearch (przy jednoczesnym zachowaniu wartości *Numeric* w bazie danych):

```
HistoryProvider=elasticsearch;value_types="str,log,text,json",url=http://localhost:9200
```

Jeśli używasz indeksów opartych na dacie, dodaj `date_index=1` do parametru:

```
HistoryProvider=elasticsearch;value_types="str,log,text,json",url=http://localhost:9200,date_index=1
```

Po wprowadzeniu zmian uruchom ponownie serwer Zabbix:

```
systemctl restart zabbix-server
```

Konfigurowanie frontend Zabbix

W pliku konfiguracyjnym frontend Zabbix (`zabbix.conf.php`) ustaw zmienną `$HISTORY_PROVIDERS` tak, aby odpowiadała konfiguracji serwera:

```
$HISTORY_PROVIDERS[] = [
  'types' => ['str','log','text','json'],
  'provider' => 'elasticsearch',
  'url' => 'http://localhost:9200'
];
```

Rozwiązywanie problemów

Poniższe kroki mogą pomóc w rozwiązywaniu problemów z konfiguracją Elasticsearch:

1. Sprawdź dzienniki Elasticsearch lub serwera Zabbix pod kątem błędów.
2. Aby zidentyfikować wolne zapytania, użyj opcji `log_slow_queries` w parametrze konfiguracji serwera Zabbix `HistoryProvider`.
3. Sprawdź, czy Elasticsearch zezwala na dostęp z serwera Zabbix i frontend Zabbix.
4. Sprawdź, czy `auto_create_index` jest włączone:

```
curl -X GET \
 "http://localhost:9200/_cluster/settings?include_defaults=true&filter_path=**.auto_create_index"

#### {"defaults": {"action": {"auto_create_index": "false"} } }
```

Aby je włączyć, wyślij następujące żądanie:

```
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. Sprawdź, czy mapowania, szablony i potoki ingest są poprawne, wysyłając żądania GET do ich odpowiednich adresów URL:

```
curl -X GET http://localhost:9200/json
curl -X GET http://localhost:9200/_template/json*
curl -X GET http://localhost:9200/_ingest/pipeline/json*
```

Otrzymane odpowiedzi możesz porównać z oczekiwanymi odpowiedziami w dokumentacji API Elasticsearch.

6. Sprawdź, czy jakiekolwiek shardy nie znajdują się w stanie awarii; ponowne uruchomienie Elasticsearch może to rozwiązać.
7. Wykonaj zapytanie do Elasticsearch, aby sprawdzić, czy dane zebrane przez Zabbix są przechowywane, na przykład:

```
curl 'http://localhost:9200/json/_search' \
 -H 'Content-Type: application/json' \
 -d '{
```

```
  "query": {
    "term": {
      "itemid": 42269
    }
  }
}'
```

8. Jeśli musisz zresetować konfigurację Elasticsearch i zacząć od nowa, możesz usunąć wszystkie indeksy, szablony i potoki ingest:

```
curl -X DELETE "http://localhost:9200/_all"
curl -X DELETE "http://localhost:9200/_template/*"
curl -X DELETE "http://localhost:9200/_ingest/pipeline/*"
```

Konfiguracja ClickHouse

Zabbix może przechowywać dane historyczne w ClickHouse jako alternatywę dla relacyjnej bazy danych.

Note:
ClickHouse może być również używany jako źródło danych telemetrycznych do monitorowania wydajności aplikacji (APM); szczegółowe informacje można znaleźć w sekcji *Administracja* > *Źródło danych*.

Niniejszy przewodnik obejmuje konfigurację obsługiwanych wersji ClickHouse. Jeśli używasz innej wersji, niektóre funkcje mogą nie działać zgodnie z oczekiwaniami.

ClickHouse może przechowywać następujące typy wartości:

Typ wartości pozycji	Tabela bazy danych	Typ ClickHouse
Liczbowa (bez znaku)	history_uint	uint
Liczbowa (zmiennoprzecinkowa)	history	dbl
Znakowa	history_str	str
Dziennik	history_log	log
Tekstowa	history_text	text
Binarna	history_bin	nieobsługiwany przez Zabbix
JSON	history_json	json

Note:
ClickHouse nie akceptuje tablic JSON. Wartość JSON musi być pojedynczym obiektem lub zestawem obiektów. Ponadto ClickHouse obsługuje klucze JSON o wartości NULL tak samo jak brakujące klucze.

Ważne uwagi

- Housekeeper nie usuwa danych z ClickHouse. Aby kontrolować, jak długo dane są przechowywane, skonfiguruj okres przechowywania danych ClickHouse w ClickHouse.
- Zabbix **nie** oblicza ani nie przechowuje trendów w ClickHouse. Rozważ wydłużenie okresu przechowywania historii, aby zachować starsze dane.
- Jeśli chcesz przeprowadzić migrację danych historii z istniejącej bazy danych Zabbix (MySQL lub PostgreSQL) do ClickHouse, zobacz skrypty migracji schematu ClickHouse i historii.
- ClickHouse nie jest obsługiwany jako magazyn historii dla proxy Zabbix, ale może być używany jako źródło danych telemetrycznych dla proxy Zabbix z włączonym APM.
- Jeśli masz klaster ClickHouse, zobacz Konfigurowanie klastrów ClickHouse.

Konfigurowanie ClickHouse

Należy utworzyć i skonfigurować bazę danych Zabbix oraz użytkownika, a także zaimportować schemat bazy danych.

Ten przewodnik zawiera instrukcje dotyczące instalacji ClickHouse za pomocą Docker lub pakietów.

Docker

1. Utwórz i skonfiguruj bazę danych Zabbix oraz użytkownika podczas uruchamiania kontenera 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. Potwierdź, że ClickHouse działa i że możesz się z nim połączyć:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --query "SELECT version()"

#### 26.4.4.38
```

3. Zaimportuj schemat bazy danych za pomocą skryptu `history_all.sh` z katalogu Zabbix:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_all.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123
```

Note:
Domyślny okres przechowywania danych (Time-To-Live, TTL) dla ClickHouse wynosi 31 dni. Aby go zmienić, użyj opcji `--ttl` podczas importowania schematu bazy danych (np. `--ttl 604800` dla 7 dni) albo skonfiguruj go później.

4. Sprawdź, czy tabele zostały utworzone:

```
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
```

Pakiety

1. Uruchom ClickHouse:

```
sudo service clickhouse-server start
```

2. Utwórz i skonfiguruj bazę danych oraz użytkownika 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. Zaimportuj schemat bazy danych, używając skryptu `history_all.sh` z katalogu Zabbix:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_all.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123
```

Note:
Domyślny okres przechowywania danych (Time-To-Live, TTL) dla ClickHouse wynosi 31 dni. Aby go zmienić, użyj opcji `--ttl` podczas importowania schematu bazy danych (np. `--ttl 604800` dla 7 dni) albo skonfiguruj go później.

4. Sprawdź, czy tabele zostały utworzone:

```
clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "SHOW TABLES FROM zabbix"

#### history
#### history_json
#### history_log
#### history_str
#### history_text
#### history_uint
```

Konfigurowanie serwera Zabbix

W pliku konfiguracyjnym serwera Zabbix (`zabbix_server.conf`) ustaw parametr `HistoryProvider`.

Na przykład, aby przechowywać wszystkie obsługiwane typy wartości w ClickHouse:

```
HistoryProvider=clickhouse;value_types="uint,dbl,str,log,text,json",url=http://localhost:8123,db=zabbix,us
```

Po wprowadzeniu zmian uruchom ponownie serwer Zabbix:

```
systemctl restart zabbix-server
```

Konfigurowanie frontend Zabbix

W pliku konfiguracyjnym frontend Zabbix (`zabbix.conf.php`) ustaw zmienną `$HISTORY_PROVIDERS` zgodnie z konfiguracją serwera:

```
$HISTORY_PROVIDERS[] = [
  'types' => ['uint','dbl','str','log','text','json'],
  'provider' => 'clickhouse',
  'url' => 'http://localhost:8123',
  'db' => 'zabbix',
  'username' => 'zabbix',
  'password' => '<password>'
];
```

Dodatkowa konfiguracja

Poniższe kroki są opcjonalne. Nie są potrzebne do podstawowej konfiguracji.

Konfigurowanie okresu przechowywania danych ClickHouse

Domyślny okres przechowywania danych (Time-To-Live, TTL) dla ClickHouse wynosi 31 dni. Aby go zmienić, uruchom poniższe polecenia.

Poniższe przykłady używają Dockera. Jeśli zainstalowano ClickHouse za pomocą pakietów, uruchom zapytania bezpośrednio w kliencie ClickHouse.

1. Zmień tabelę (zastąp `history_json` i 3600 wartościami, których potrzebujesz):

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "ALTER TABLE zabbix.history_json MODIFY TTL clock_ns + toIntervalSecond(3600)"
```

2. Zastosuj zmiany natychmiast:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "OPTIMIZE TABLE zabbix.history_json MATERIALIZE TTL"
```

3. Sprawdź, czy okres przechowywania danych został zmieniony:

```
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. Uruchom ponownie serwer Zabbix, aby odświeżyć okres przechowywania danych w *Administration* > *Housekeeping*:

```
systemctl restart zabbix-server
```

Konfigurowanie klastrów ClickHouse

Ta sekcja zawiera kroki konfiguracji klastrów ClickHouse z replikacją i shardowaniem.

Poniższe kroki wykorzystują Docker oraz ClickHouse `cluster_2S_2R` z dokumentacji ClickHouse (uruchomiony na ClickHouse 26.4.4.38 z ClickHouse Keeper).

Przed rozpoczęciem zaleca się umieszczenie load balancera przed klastrem ClickHouse dla Zabbix. Konfiguracja serwera Zabbix i frontend obsługuje tylko jeden adres URL ClickHouse, więc może wskazywać tylko na jeden węzeł. Jeśli ten węzeł przestanie działać, load balancer przekieruje Zabbix do sprawnego węzła, korzystając z tego samego adresu URL.

1. Utwórz bazę danych 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:
Klauzula `ON CLUSTER` obejmuje tylko te węzły, które są częścią klastra w momencie uruchomienia polecenia. Jeśli później dodasz węzły do klastra, uruchom to polecenie ponownie.

2. Utwórz i skonfiguruj użytkownika 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. Zaimportuj schemat bazy danych, używając skryptu history_all.sh z katalogu Zabbix:

```
./usr/share/zabbix/sql-scripts/clickhouse/history_all.sh \
  --user zabbix \
  --password <password> \
  --db zabbix \
  --server http://localhost:8123 \
  --engine "ReplicatedMergeTree()"
```

Note:
Domyślny okres przechowywania danych (Time-To-Live, TTL) w ClickHouse wynosi 31 dni. Aby go zmienić, użyj opcji `--ttl` podczas importowania schematu bazy danych (np. `--ttl 604800` dla 7 dni) lub skonfiguruj go później.

Note:
Jeśli klaster ClickHouse wymaga jawnych ścieżek replik (na przykład gdy konfiguracja serwera ClickHouse nie ustawia domyślnych ścieżek dla tabel replikowanych), uruchom skrypty schematu (`history_schema.sh`, `history_uint_schema.sh` itd.) osobno i ustaw inną ścieżkę dla każdej tabeli:

```
./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}')"
```

Więcej informacji o replikacji ClickHouse znajdziesz w dokumentacji ClickHouse: Replicating data oraz Replicated* table engines.

4. Skonfiguruj shardowanie dla każdego typu tabeli. Zmień nazwy tabel i utwórz tabelę `Distributed` o oryginalnej nazwie, wskazującą na przemianowaną tabelę. To właśnie z tej tabeli `Distributed` Zabbix faktycznie odczytuje i do niej zapisuje dane; tabela automatycznie rozdziela dane między shardami. Na przykład dla tabeli `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:
Więcej informacji o shardowaniu ClickHouse znajdziesz w dokumentacji ClickHouse w sekcji Distributed table engine.

5. Skonfiguruj serwer Zabbix.

6. Skonfiguruj frontend Zabbix.

7. Sprawdź, czy dane Zabbix są rozłożone między shardami. Liczba wierszy w `zabbix.history_local` może się zmieniać między kolejnymi sprawdzeniami, ponieważ serwer Zabbix stale wstawia nowe dane. Uruchom to polecenie na każdym węźle:

```
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. Sprawdź, czy replikacja jest zsynchronizowana (`active_replicas` powinno odpowiadać `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
```

Rozwiązywanie problemów

Poniższe kroki mogą pomóc w rozwiązywaniu problemów z konfiguracją ClickHouse:

1. Sprawdź dzienniki ClickHouse lub serwera Zabbix pod kątem błędów.
2. Aby zidentyfikować wolne zapytania, użyj opcji `log_slow_queries` w parametrze konfiguracji serwera Zabbix `HistoryProvider`.
3. Sprawdź, czy ClickHouse zezwala na dostęp z serwera Zabbix i frontend Zabbix.
4. Wykonaj zapytanie do ClickHouse, aby sprawdzić, czy dane zbierane przez Zabbix są zapisywane, na przykład:

```
sudo docker exec -it clickhouse \
  clickhouse-client \
  --user zabbix \
  --password <password> \
  --query "SELECT * FROM zabbix.history_uint WHERE itemid = 42269"
```

Uwagi dotyczące konfiguracji NGINX dla Zabbix zależne od dystrybucji

RHEL

W systemach RHEL 8, 9 i 10 NGINX jest dostępny w AppStream — nie są wymagane dodatkowe repozytoria. Wystarczy włączyć moduł i zainstalować NGINX za pomocą:

```
dnf module list nginx
dnf -y install nginx
```

W systemie RHEL 7 NGINX można zainstalować z repozytorium CentOS 7 Extras (jeśli używany jest CentOS), wykonując:

```
yum -y install nginx
```

Jeśli preferujesz najnowsze kompilacje upstream, skonfiguruj oficjalne repozytorium NGINX w pliku `/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

W SUSE Linux Enterprise Server 15 należy skonfigurować `php-fpm` (ścieżka do pliku konfiguracyjnego może się nieznacznie różnić w zależności od dodatku 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
```

Konfiguracja SAML z Microsoft Entra ID

Przegląd

Ta sekcja zawiera wytyczne dotyczące konfigurowania logowania jednokrotnego i konfiguracja kont użytkowników w Zabbix z Microsoft Entra ID (dawniej Microsoft Azure Active Directory) przy użyciu uwierzytelniania SAML 2.0.

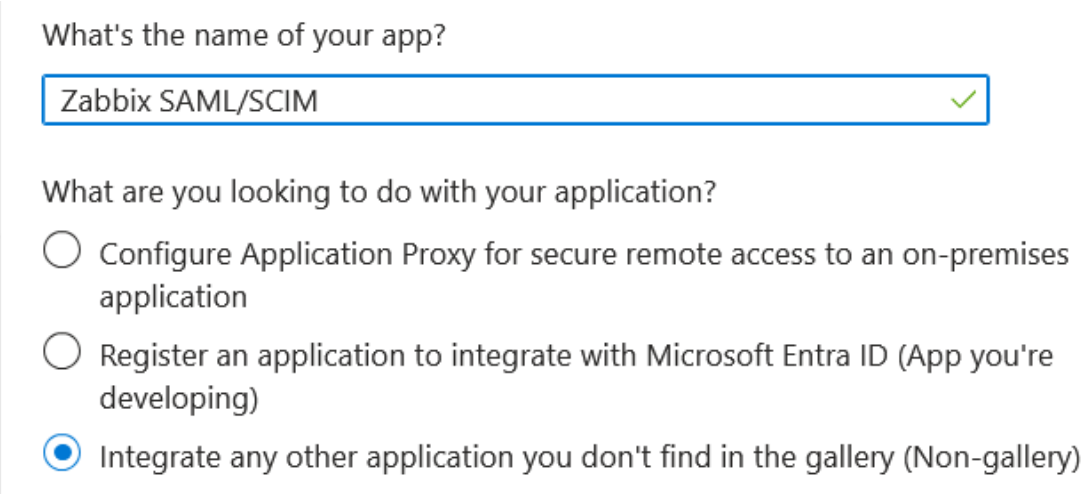
Konfiguracja ID Microsoft Entra

Tworzenie aplikacji

1. Zaloguj się do centrum administracyjnego Microsoft Entra w Microsoft Entra ID. Do celów testowych możesz utworzyć bezpłatne konto próbne w Microsoft Entra ID.

2. W centrum administracyjnym Microsoft Entra wybierz *Applications* > *Enterprise applications* > *New application* > *Create your own application*.

3. Dodaj nazwę swojej aplikacji i wybierz opcję *Integrate any other application...*. Następnie kliknij *Create*.

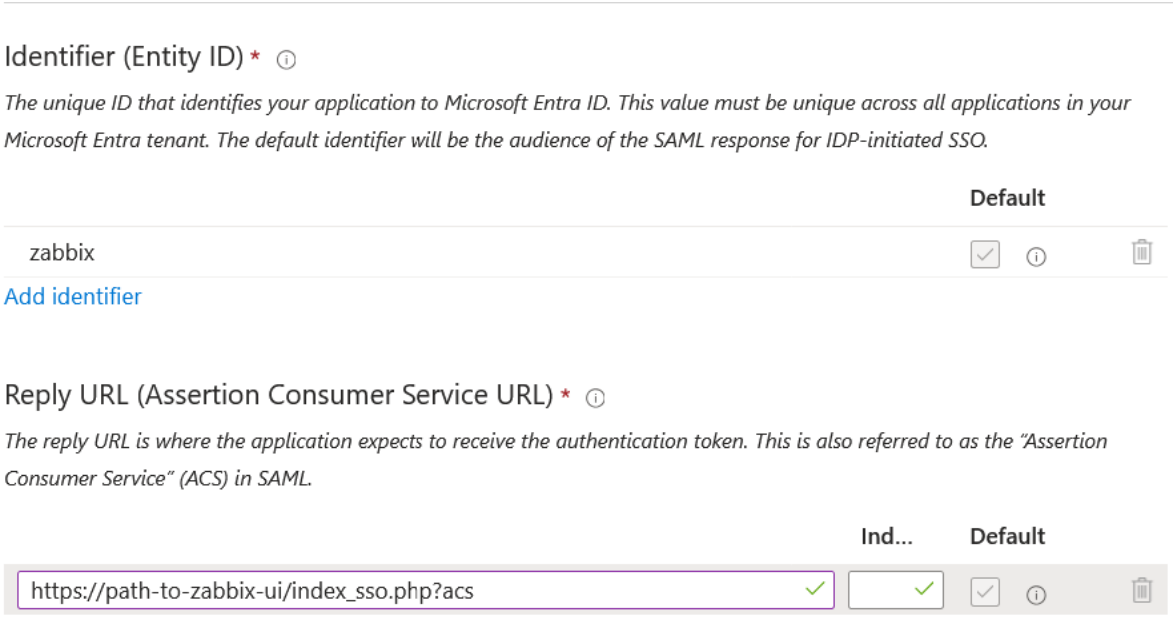


Konfigurowanie logowania jednokrotnego

1. Na stronie aplikacji przejdź do *Set up single sign on* i kliknij *Get started*. Następnie wybierz *SAML*.

2. Edytuj *Basic SAML Configuration*:

- W polu *Identifier (Entity ID)* ustaw unikatową nazwę identyfikującą aplikację w Microsoft Entra ID, na przykład `zabbix`;
- W polu *Reply URL (Assertion Consumer Service URL)* ustaw punkt końcowy logowania jednokrotnego Zabbix: `https://<path-to-zabb`



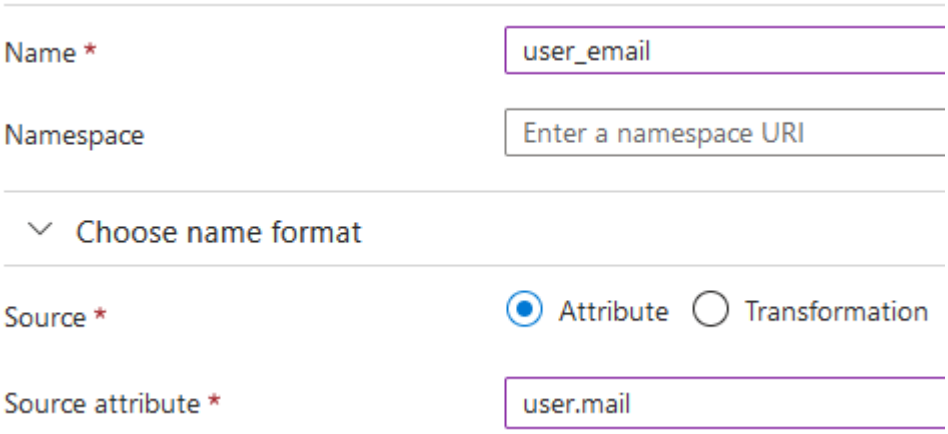
Pamiętaj, że wymagany jest protokół `https`. Aby używać tej funkcji z Zabbix, dodaj następujący wiersz do pliku konfiguracyjnego frontendu:

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

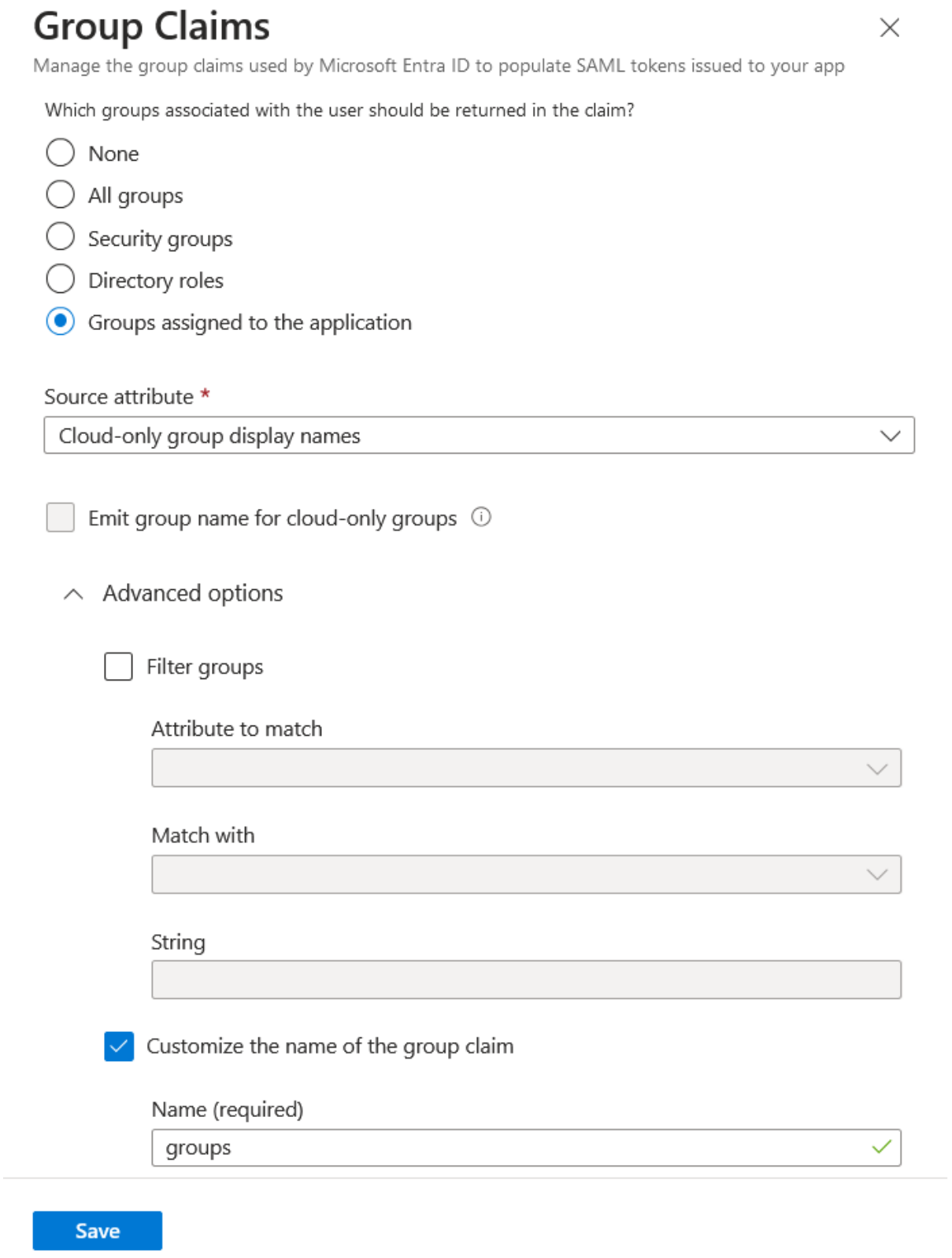
3. Edytuj *Attributes & Claims*. Musisz dodać wszystkie atrybuty, które chcesz przekazywać do Zabbix (user_name, user_lastname, user_email, user_mobile, groups).

Nazwy atrybutów są dowolne. Można używać innych nazw atrybutów, jednak muszą one odpowiadać odpowiednim wartościom pól w ustawieniach SAML Zabbix.

- Kliknij *Add new claim*, aby dodać atrybut:



- Kliknij *Add a group claim*, aby dodać atrybut służący do przekazywania grup do Zabbix:



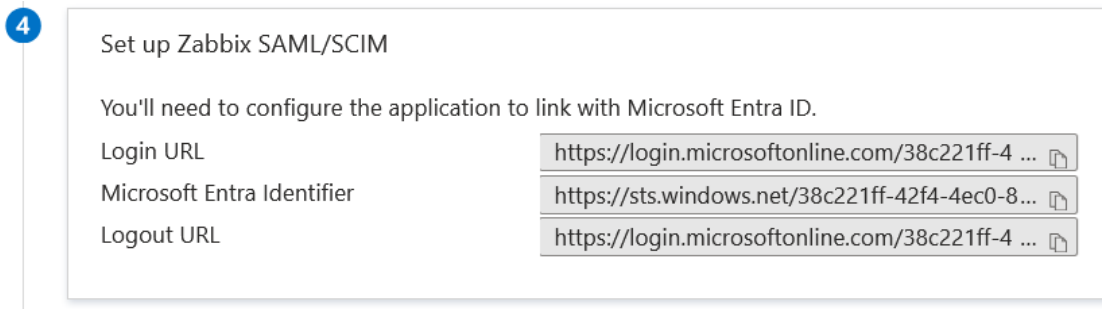
W przypadku tego oświadczenia ważne jest, aby za pomocą wybranego pola *Source attribute* do Zabbix były przekazywane nazwy grup, a nie ich identyfikatory. W przeciwnym razie automatyczne tworzenie użytkowników JIT nie będzie działać prawidłowo.

4. W sekcji *SAML Certificates* pobierz certyfikat Base64 udostępniony przez Entra ID i umieść go w katalogu `conf/certs` instalacji frontendu Zabbix (lokalizacja może się różnić w zależności od systemu operacyjnego i źródła instalacji Zabbix).

Ustaw uprawnienia 644, uruchamiając:

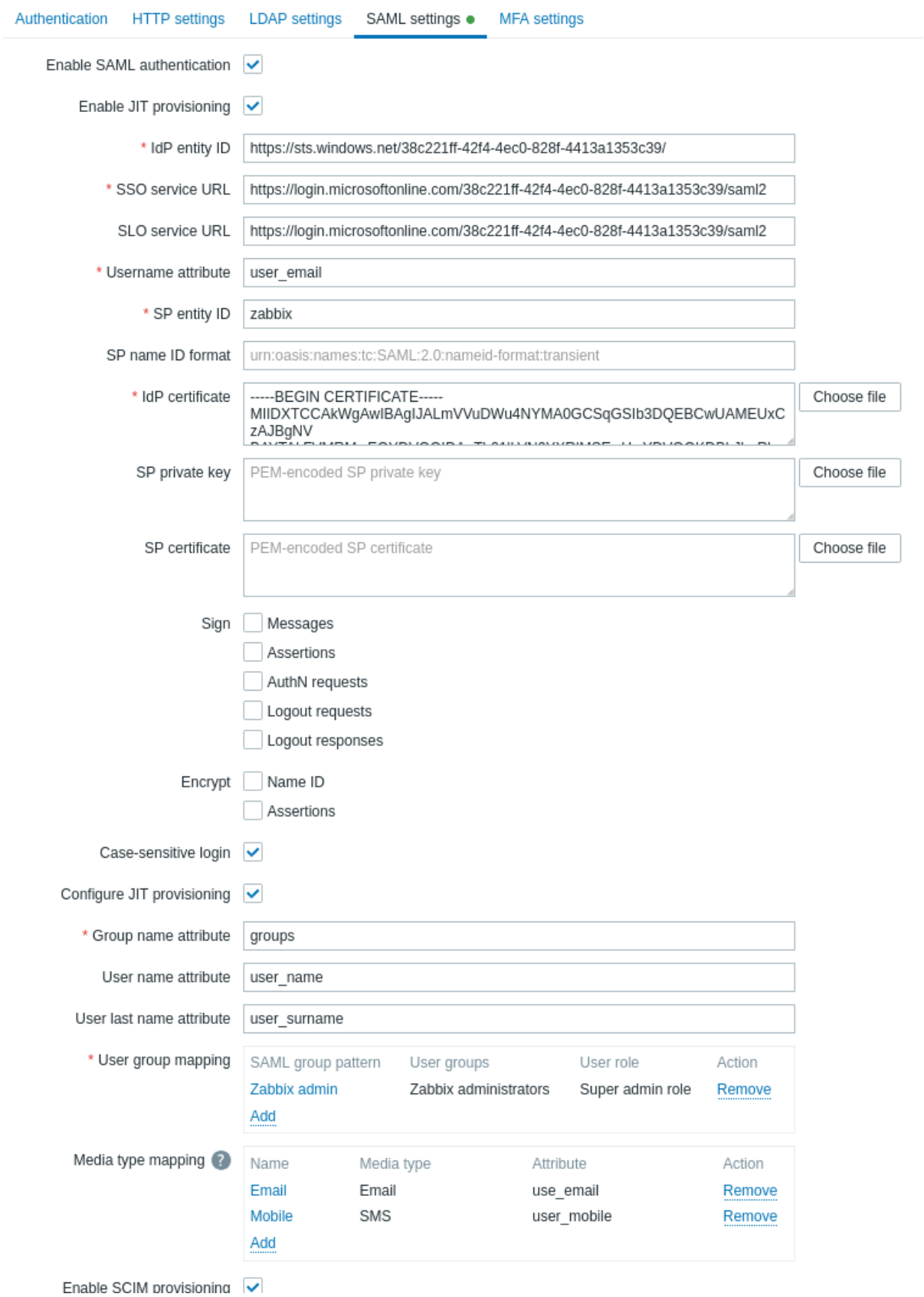
```
chmod 644 entra.cer
```

5. Użyj wartości z sekcji *Set up <your app name>* w Entra ID, aby skonfigurować uwierzytelnianie SAML w Zabbix (zobacz następną sekcję):



Konfiguracja Zabbix

1. W Zabbix przejdź do ustawień SAML i uzupełnij opcje konfiguracji na podstawie konfiguracji Entra ID:



Pole Zabbix	Pole konfiguracji w Entra ID	Przykładowa wartość
IdP entity ID	Microsoft Entra identifier	
SSO service URL	Login URL	
SLO service URL	Logout URL	
SP entity ID	Identifier (Entity ID)	
Username attribute	Custom attribute (claim)	`user_email`
Group name attribute	Custom attribute (claim)	`groups`
User name attribute	Custom attribute (claim)	`user_name`

Pole Zabbix	Pole konfiguracji w Entra ID	Przykładowa wartość
User last name attribute	Custom attribute (claim)	`user_lastname`

2. Dodaj certyfikat Base64 dostarczony przez Entra ID.

Jeśli w pliku konfiguracji frontendu (`zabbix.conf.php`) ustawiono `$SSO['CERT_STORAGE'] = 'database'`, podczas konfiguracji SAML możesz wkleić tekst certyfikatu lub przesłać plik certyfikatu w frontendzie — pliki w systemie plików nie są wymagane.

Jeśli w pliku konfiguracji ustawiono `$SSO['CERT_STORAGE'] = 'file'`, certyfikat musi być dostępny w systemie plików (domyślnie w `ui/conf/certs` lub w ścieżce skonfigurowanej w pliku konfiguracji), a frontend nie będzie przechowywać certyfikatów w bazie danych. Pamiętaj, że jeśli `$SSO['CERT_STORAGE']` nie jest ustawione lub jest zakomentowane, zakładane jest przechowywanie w pliku, a certyfikaty są odczytywane z `ui/conf/certs`.

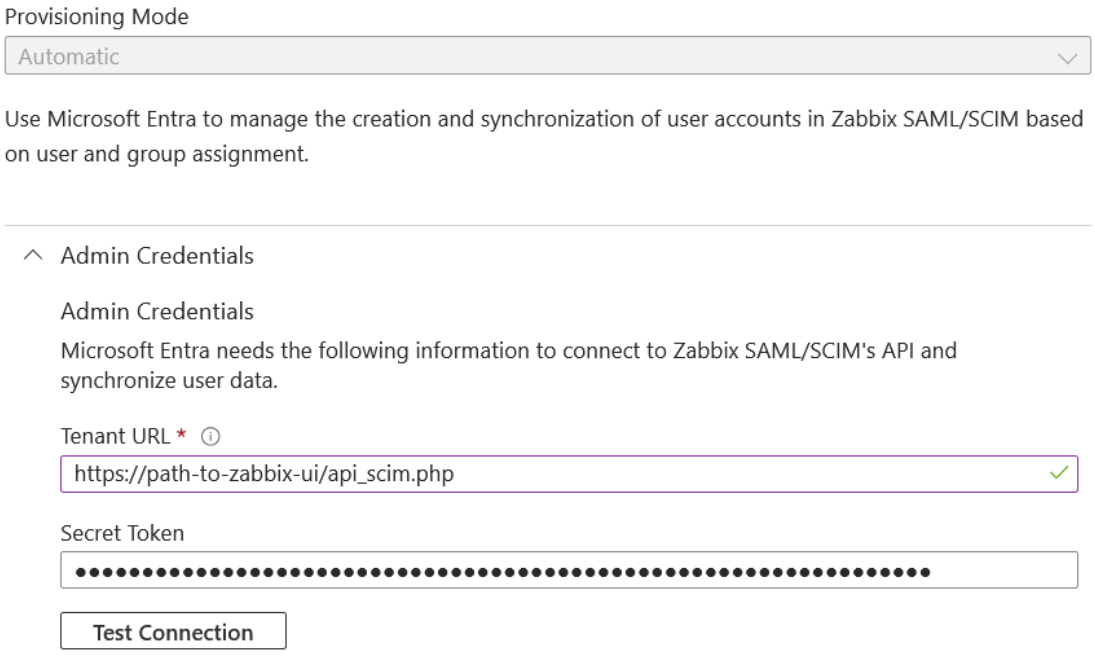
Należy również skonfigurować mapowanie grup użytkowników. Mapowanie mediów jest opcjonalne.

3. Kliknij przycisk *Update*, aby zapisać te ustawienia.

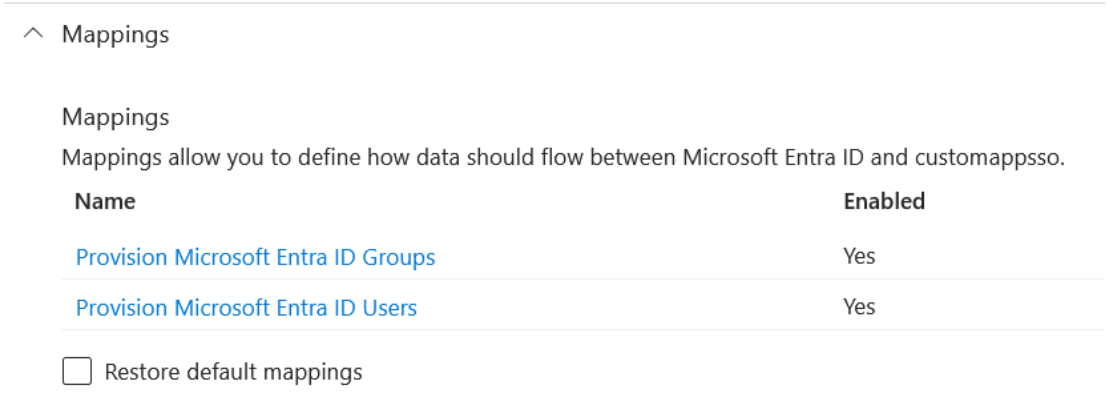
Provisioning użytkowników SCIM

1. Na stronie aplikacji Entra ID, w menu głównym otwórz stronę Provisioning. Kliknij *Get started*, a następnie wybierz tryb Automatic provisioning:

- W *Tenant URL* ustaw następującą wartość: `https://<path-to-zabbix-ui>/api_scim.php`
- W *Secret token* wprowadź token API Zabbix z uprawnieniami Super admin.
- Kliknij *Test connection*, aby sprawdzić, czy połączenie zostało nawiązane.

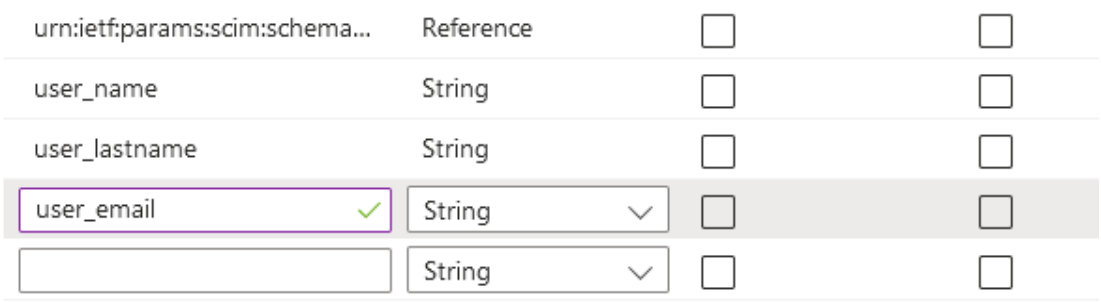


2. Teraz możesz dodać wszystkie atrybuty, które będą przekazywane przez SCIM do Zabbix. Aby to zrobić, kliknij *Mappings*, a następnie *Provision Microsoft Entra ID Users*.



Na dole listy Attribute Mapping włącz *Show advanced options*, a następnie kliknij *Edit attribute list for customappsso*.

Na dole listy atrybutów dodaj własne atrybuty z typem 'String':



Zapisz listę.

3. Teraz możesz dodać mapowania dla dodanych atrybutów. Na dole listy Attribute Mapping kliknij *Add New Mapping* i utwórz mapowania pokazane poniżej:



Gdy wszystkie mapowania zostaną dodane, zapisz listę mapowań.



4. Jako warunek wstępny provisioning użytkowników do Zabbix, musisz mieć skonfigurowanych użytkowników i grupy w Entra ID.

Aby to zrobić, przejdź do *Microsoft Entra admin center*, a następnie dodaj użytkowników/grupy na odpowiednich stronach Users i Groups.

5. Gdy użytkownicy i grupy zostaną utworzeni w Entra ID, możesz przejść do menu *Users and groups* swojej aplikacji i dodać ich do aplikacji.

6. Przejdź do menu *Provisioning* swojej aplikacji i kliknij *Start provisioning*, aby rozpocząć provisioning użytkowników do Zabbix.

Należy pamiętać, że żądanie PATCH Users w Entra ID nie obsługuje zmian w media.

Podpisywanie żądań uwierzytelniania

Można skonfigurować Entra ID tak, aby weryfikował podpis podpisanych żądań uwierzytelniania.

Aby to zadziałało, utwórz klucze publiczny i prywatny:

```
openssl req -x509 -newkey rsa:4096 -keyout /usr/share/zabbix/conf/certs/request-sign.key -out /usr/share/z
```

Przypisz uprawnienia:

```
chown apache /usr/share/zabbix/conf/certs/request-sign.key
chmod 400 /usr/share/zabbix/conf/certs/request-sign.key
```

Zaktualizuj konfigurację frontendu Zabbix, dodając:

```
$SSO['SP_KEY'] = 'conf/certs/request-sign.key';
$SSO['SP_CERT'] = 'conf/certs/request-sign.crt';
```

Rozwiązywanie problemów

Problemy z uwierzytelnianiem mogą wystąpić w przeglądarce Microsoft Edge, gdy użytkownik próbujący zalogować się do Zabbix za pośrednictwem SAML jest już zalogowany przy użyciu profilu Microsoft Edge. Oznaką takiego problemu może być możliwość zalogowania się do Zabbix za pomocą przeglądarki Microsoft Edge w trybie prywatnym.

Aby uniknąć problemów z uwierzytelnianiem w tym przypadku, może być konieczne ustawienie parametru `requestedAuthnContext` na `false` w pliku konfiguracji frontendu Zabbix.

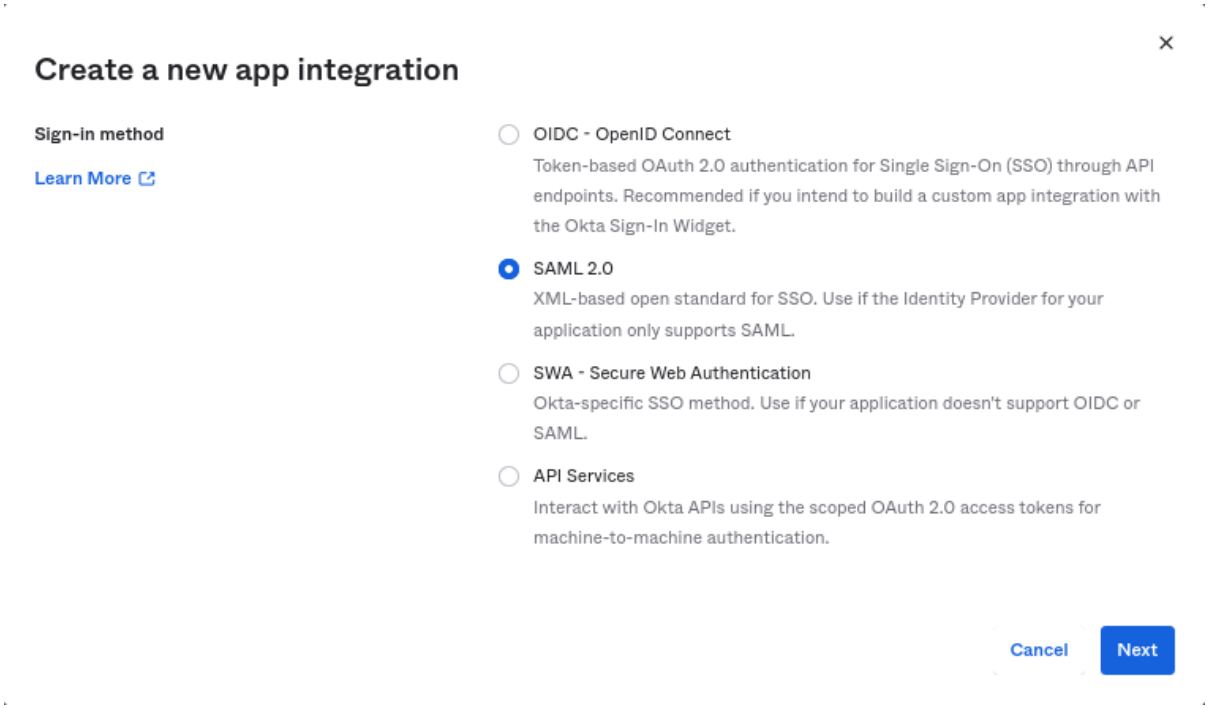
```
$SSO['SETTINGS'] = [
    'security' => [
        'requestedAuthnContext' => false
    ]
];
```

Konfiguracja SAML z Okta

Ta sekcja zawiera wskazówki dotyczące konfiguracji Okta, aby włączyć uwierzytelnianie SAML 2.0 oraz provisioning użytkowników dla Zabbix.

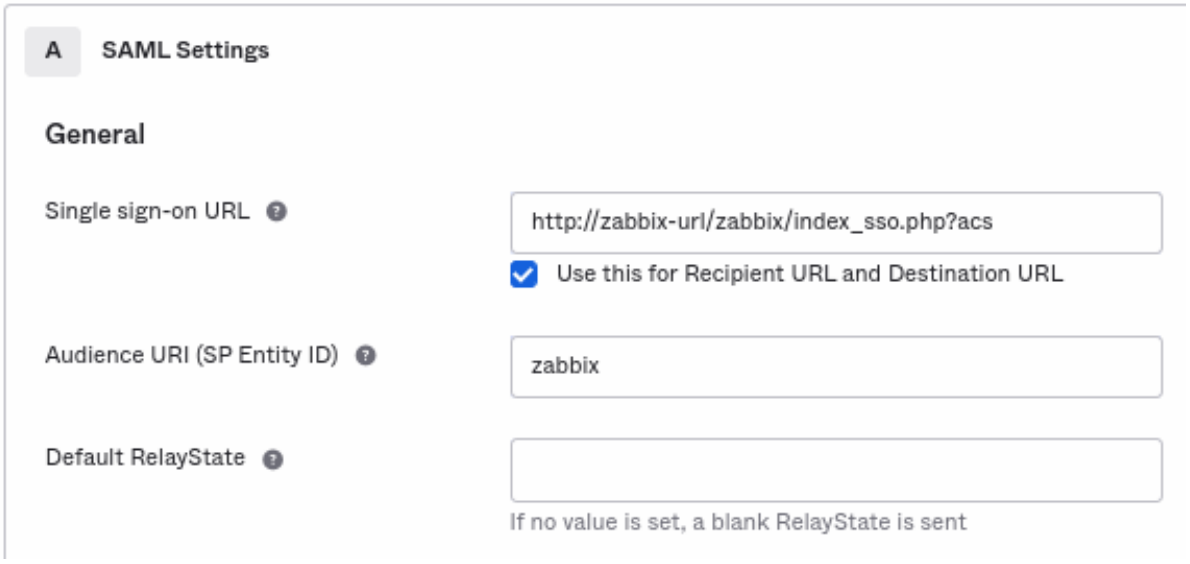
Konfiguracja Okta

1. Przejdź do https://developer.okta.com/signup/ i zarejestruj się / zaloguj na swoje konto.
2. W interfejsie webowym Okta przejdź do *Applications* > *Applications*.
3. Kliknij *Create App Integration*.

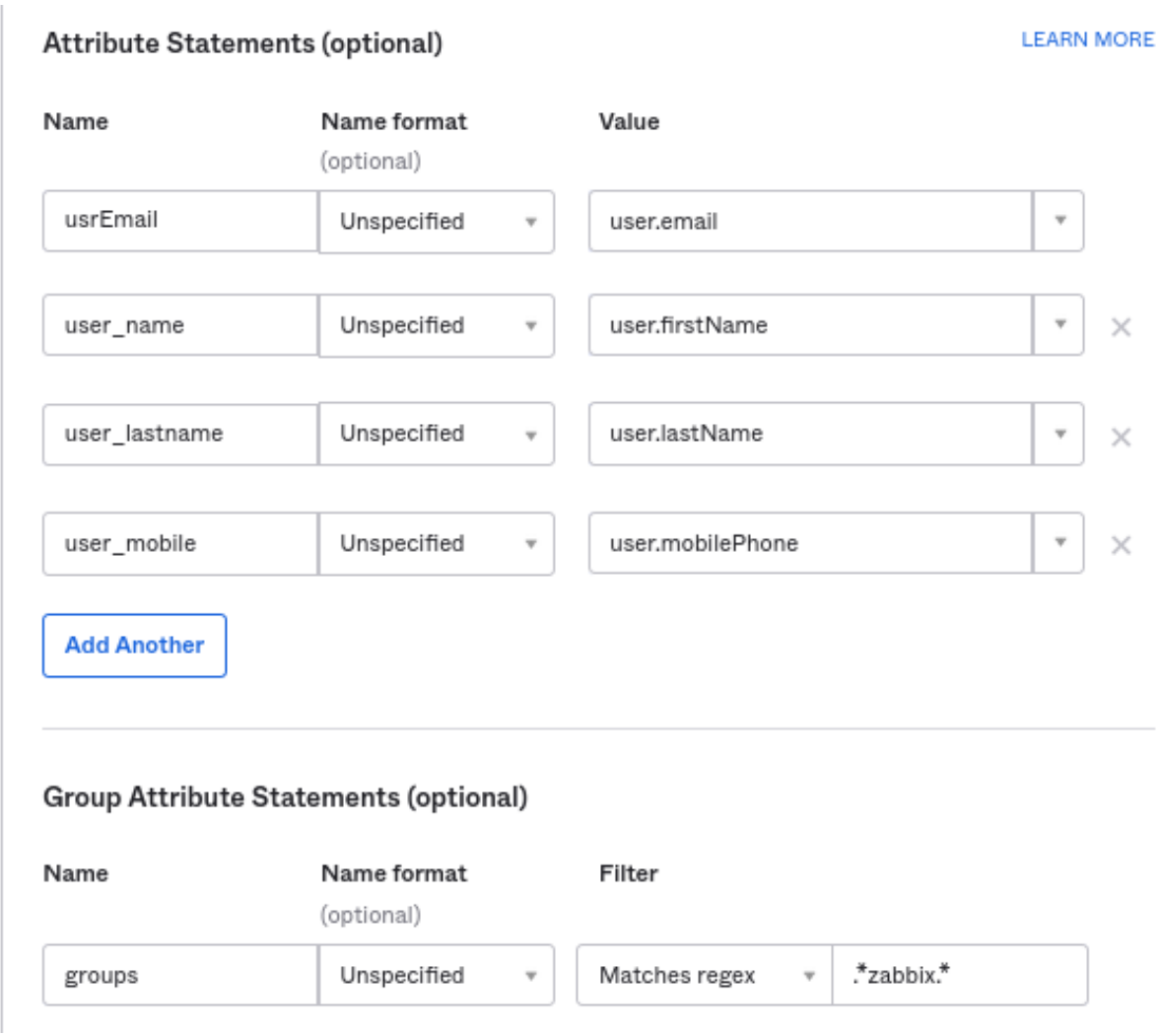


Wybierz ”SAML 2.0” jako metodę logowania i kliknij *Next*.

4. W ustawieniach ogólnych wpisz nazwę aplikacji i kliknij *Next*.
5. W konfiguracji SAML wprowadź poniższe wartości, a następnie kliknij *Next*.



- W sekcji **General** dodaj:
  - *Single sign-on URL*: `http://<your-zabbix-url>/zabbix/index_sso.php?acs<br>` Zwróć uwagę na użycie "http", a nie "https", aby parametr `acs` nie został obcięty w żądaniu. Pole wyboru *Use this for Recipient URL and Destination URL* również powinno być zaznaczone.
  - *Audience URI (SP Entity ID)*: `zabbix<br>` Zwróć uwagę, że ta wartość będzie używana w asercji SAML jako unikalny identyfikator dostawcy usług (jeśli nie będzie zgodna, operacja zostanie odrzucona). W tym polu można podać adres URL lub dowolny ciąg znaków.
  - *Default RelayState*:
 Pozostaw to pole puste; jeśli wymagane jest niestandardowe przekierowanie, można je dodać w Zabbix w ustawieniach *Users* > *Users*.
  - Uzupełnij pozostałe pola zgodnie z własnymi preferencjami.
- W sekcji **Attribute Statements/Group Attribute Statements** dodaj:



Te instrukcje atrybutów są wstawiane do asercji SAML udostępnianych Zabbix.

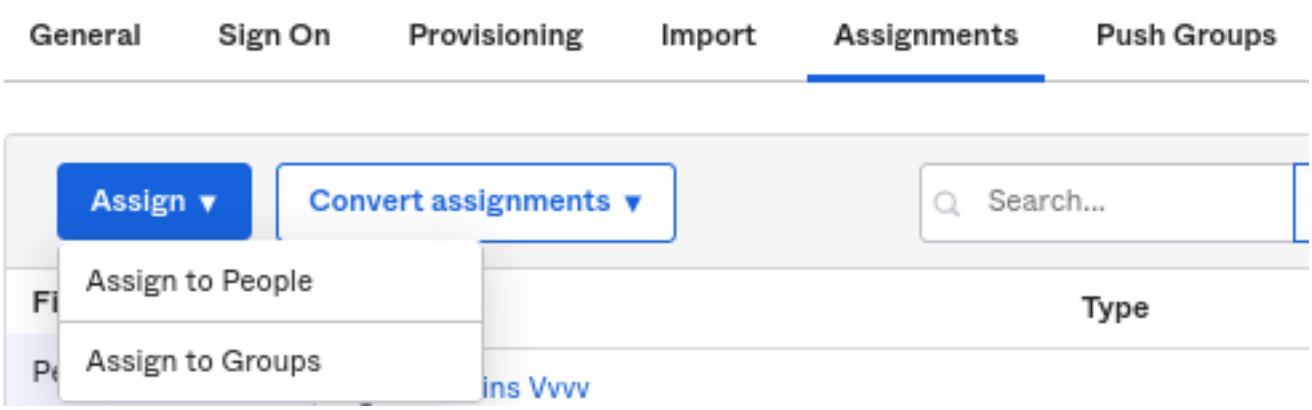
Użyte tutaj nazwy atrybutów są przykładowe i dowolne. Możesz użyć innych nazw atrybutów, jednak muszą one odpowiadać odpowiednim wartościom pól w ustawieniach SAML Zabbix.

Jeśli chcesz skonfigurować logowanie SAML do Zabbix *bez* JIT provisioning użytkowników, wymagany jest tylko atrybut email.

Note:
Jeśli planujesz używać szyfrowanego połączenia, wygeneruj prywatny i publiczny certyfikat szyfrowania, a następnie prześlij publiczny certyfikat do Okta. Formularz przesyłania certyfikatu pojawi się, gdy *Assertion Encryption* zostanie ustawione na "Encrypted" (kliknij *Show Advanced Settings*, aby znaleźć ten parametr).

6. W następnej karcie wybierz "I'm a software vendor. I'd like to integrate my app with Okta" i kliknij *Finish*.

7. Przejdź do karty "Assignments" nowo utworzonej aplikacji i kliknij przycisk *Assign*, a następnie wybierz z listy rozwijanej *Assign to People*.



8. W wyświetlonym oknie przypisz aplikację osobom, które będą używać SAML 2.0 do uwierzytelniania w Zabbix, a następnie kliknij *Save and go back*.

9. Pobierz certyfikat IdP. Nadaj mu uprawnienia 644, uruchamiając:

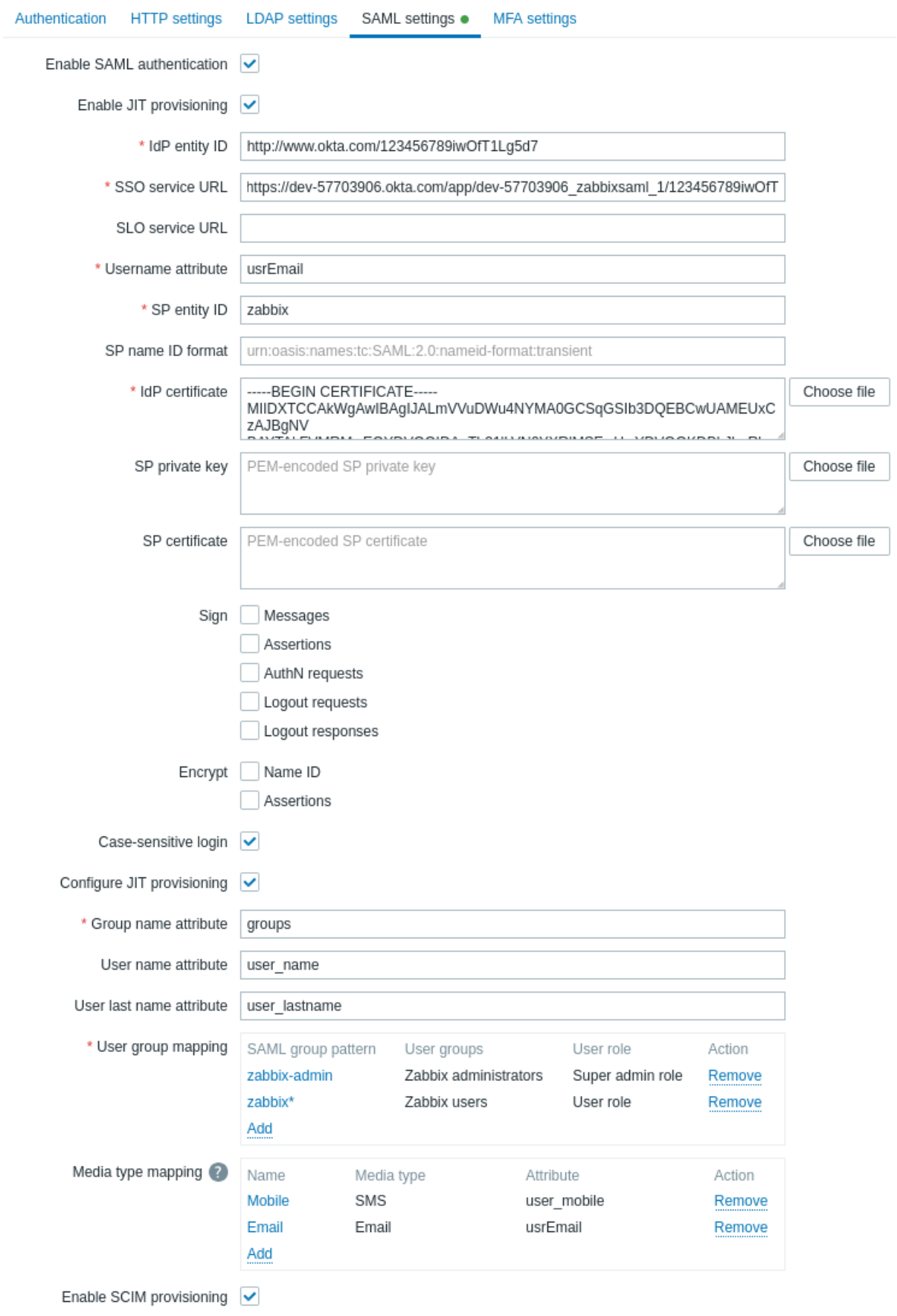
```
chmod 644 idp.crt
```

10. Przejdź do karty "Sign On" i kliknij przycisk *View Setup Instructions*.

Instrukcje konfiguracji zostaną otwarte w nowej karcie; pozostaw tę kartę otwartą podczas konfigurowania Zabbix.

Konfiguracja Zabbix

1. W Zabbix przejdź do ustawień SAML i uzupełnij opcje konfiguracji na podstawie instrukcji konfiguracji z Okta:



Pole Zabbix	Pole konfiguracji w Okta	Przykładowa wartość
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`

Pole Zabbix	Pole konfiguracji w Okta	Przykładowa wartość
User name attribute	Attribute name	`user_name`
User last name attribute	Attribute name	`user_lastname`

Należy również skonfigurować mapowanie grup użytkowników i mediów.

2. Dodaj certyfikat Base64 podany w instrukcjach konfiguracji SAML w Okta.

Jeśli w pliku konfiguracji frontendu (`zabbix.conf.php`) ustawiono `$SSO['CERT_STORAGE'] = 'database'`, możesz wkleić treść certyfikatu lub przesłać plik certyfikatu w frontendzie podczas konfiguracji SAML — pliki w systemie plików nie są wymagane.

Jeśli w pliku konfiguracji ustawiono `$SSO['CERT_STORAGE'] = 'file'`, certyfikat musi być dostępny w systemie plików (domyślnie w `ui/conf/certs` lub w ścieżce skonfigurowanej w pliku konfiguracji), a frontend nie będzie przechowywać certyfikatów w bazie danych. Jeśli `$SSO['CERT_STORAGE']` nie jest ustawione lub jest zakomentowane, używane jest przechowywanie w pliku, a certyfikaty są odczytywane z `ui/conf/certs`.

3. Jeśli w Okta dla opcji *Assertion Encryption* ustawiono wartość „Encrypted", w Zabbix należy również zaznaczyć pole wyboru „Assertions" parametru *Encrypt*.

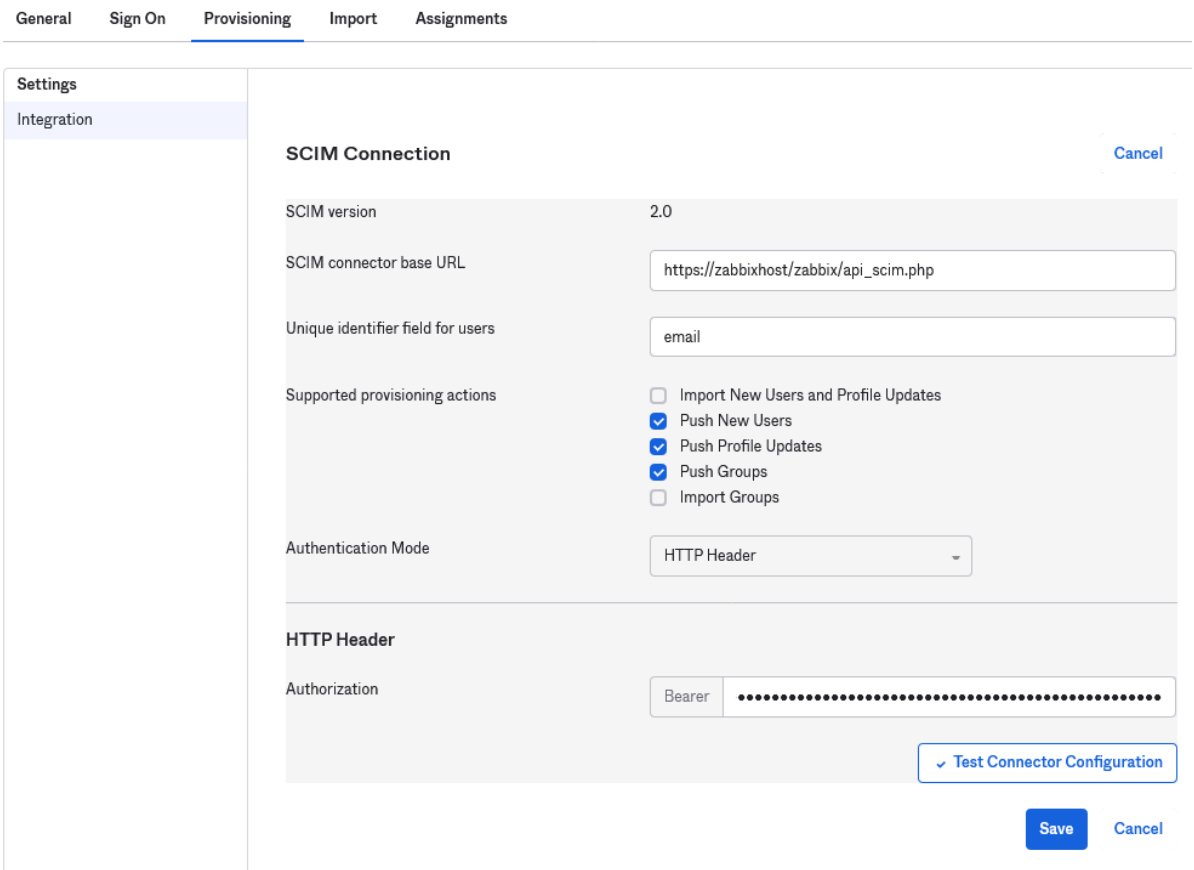
4. Kliknij przycisk *Update*, aby zapisać te ustawienia.

Inicjowanie obsługi SCIM

1. Aby włączyć inicjowanie obsługi SCIM, przejdź do sekcji *General* > *App Settings* aplikacji w Okta.

Zaznacz pole wyboru *Enable SCIM provisioning*. W rezultacie pojawi się nowa karta *Provisioning*.

2. Przejdź do karty „Provisioning", aby skonfigurować połączenie SCIM: - W polu *SCIM connector base URL* określ ścieżkę do frontendu Zabbix i dodaj do niej `api_scim.php`, na przykład:
 `https://<your-zabbix-url>/zabbix/api_scim.php` - *Unique identifier field for users*: `email` - *Authentication mode*: `HTTP header` - W polu *Authorization* wprowadź prawidłowy token API z uprawnieniami Super admin.



Attention:
Jeśli występują problemy z uwierzytelnianiem, zobacz Przekazywanie nagłówka autoryzacji.

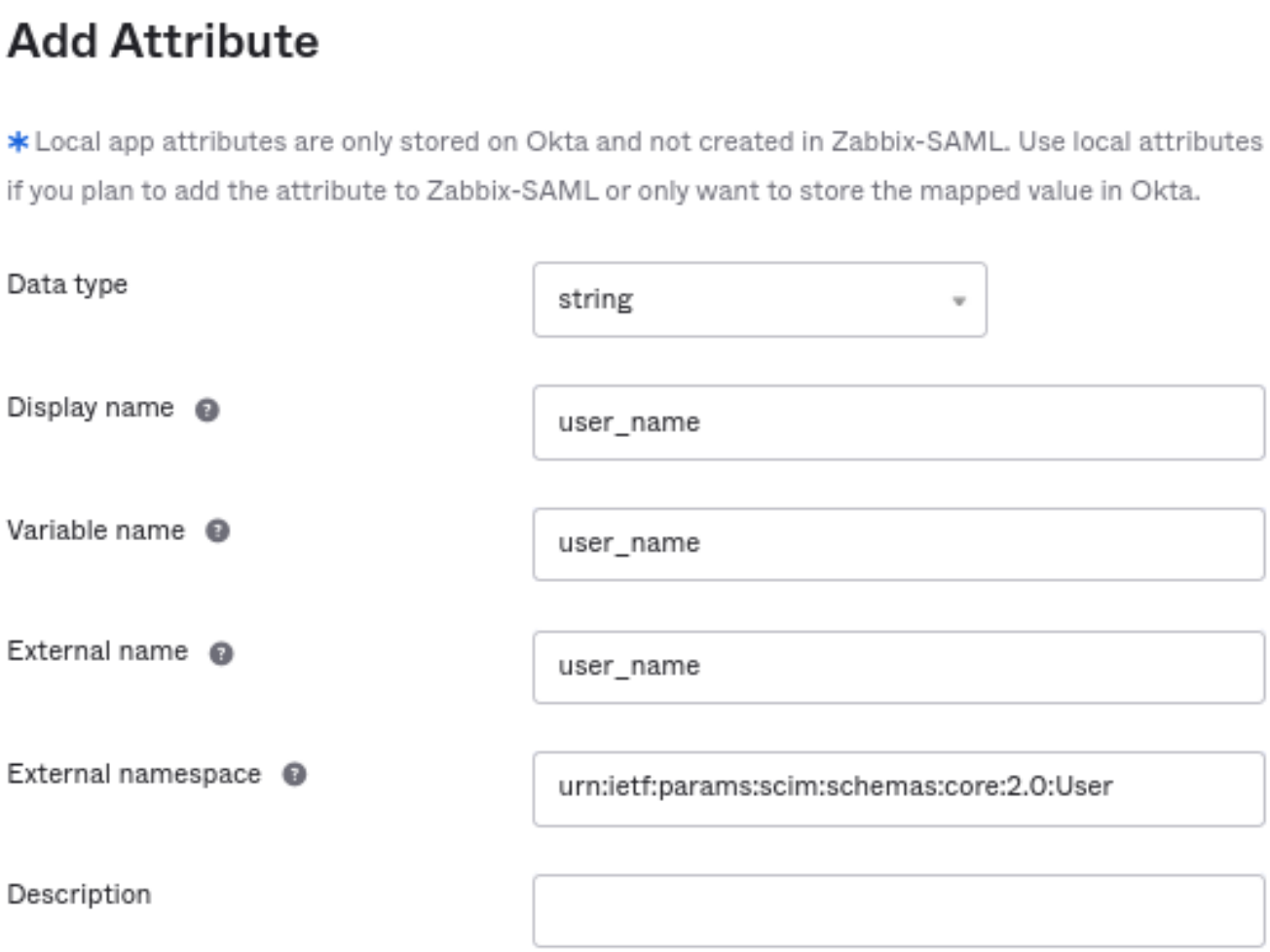
3. Kliknij *Test Connector Configuration*, aby przetestować połączenie. Jeśli wszystko jest poprawnie skonfigurowane, zostanie wyświetlony komunikat o powodzeniu.

4. W sekcji *Provisioning* > *To App* upewnij się, że zaznaczone są następujące pola wyboru: - Create Users - Update User Attributes - Deactivate Users

Dzięki temu te typy żądań będą wysyłane do Zabbix.

5. Upewnij się, że wszystkie atrybuty zdefiniowane w SAML są zdefiniowane również w SCIM. Edytor profilu aplikacji jest dostępny w sekcji *Provisioning* > *To App*. Kliknij *Go to Profile Editor*.

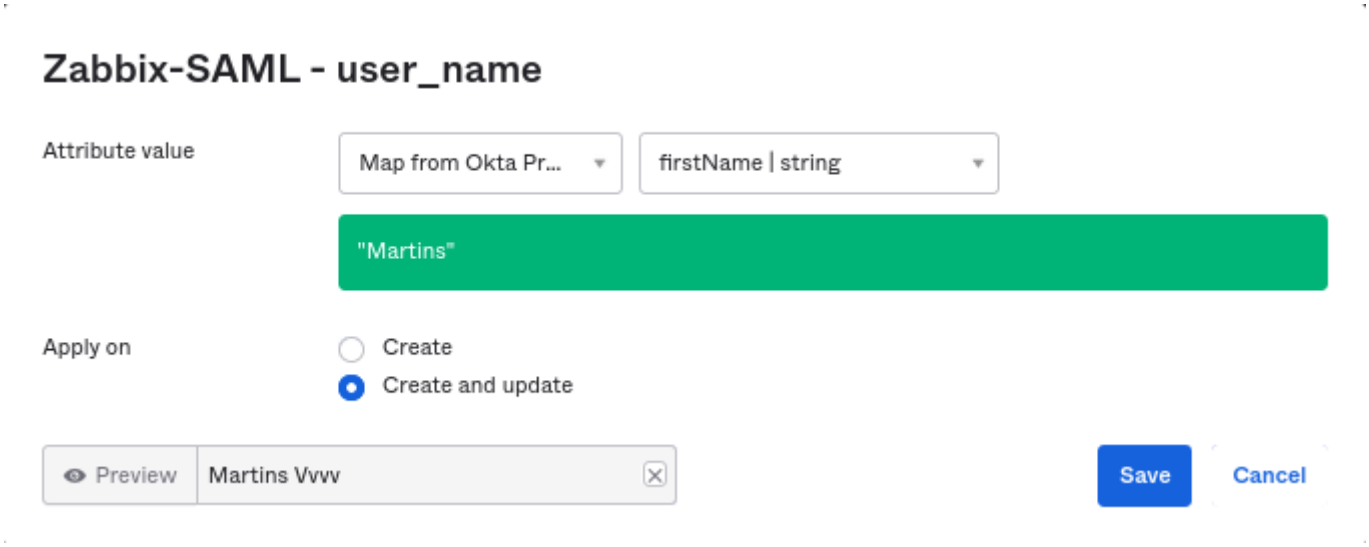
Kliknij *Add Attribute*. Wprowadź wartości w polach *Display name*, *Variable name* i *External name*, używając nazwy atrybutu SAML, na przykład `user_name`.



External namespace powinno być takie samo jak schemat użytkownika: `urn:ietf:params:scim:schemas:core:2.0:User`

6. Przejdź do sekcji *Provisioning* > *To App* > *Attribute Mappings* aplikacji. Kliknij *Show Unmapped Attributes* na dole strony. Pojawią się nowo dodane atrybuty.

7. Zmapuj każdy dodany atrybut.



8. Dodaj użytkowników na karcie „Assignments”. Użytkownicy muszą wcześniej zostać dodani w sekcji *Directory* > *People*. Wszystkie te przypisania zostaną wysłane do Zabbix jako żądania.

9. Dodaj grupy na karcie „Push Groups”. Wzorzec mapowania grup użytkowników w ustawieniach SAML Zabbix musi odpowiadać grupie określonej tutaj. Jeśli nie ma dopasowania, użytkownik nie może zostać utworzony w Zabbix.

Informacje o członkach grupy są wysyłane za każdym razem, gdy zostanie wprowadzona zmiana.

Konfiguracja SAML z OneLogin

Przegląd

Ta sekcja zawiera wytyczne dotyczące konfigurowania logowania jednokrotnego oraz aprowizacji użytkowników w Zabbix z OneLogin przy użyciu uwierzytelniania SAML 2.0.

Konfiguracja OneLogin

Tworzenie aplikacji

1. Zaloguj się na swoje konto w OneLogin. Do celów testowych możesz utworzyć bezpłatne konto deweloperskie w OneLogin.

2. W interfejsie webowym OneLogin przejdź do *Applications* > *Applications*.

3. Kliknij ”Add App” i wyszukaj odpowiednią aplikację. Wytyczne na tej stronie opierają się na przykładzie aplikacji *SCIM Provisioner with SAML (SCIM v2 Enterprise, full SAML)*.

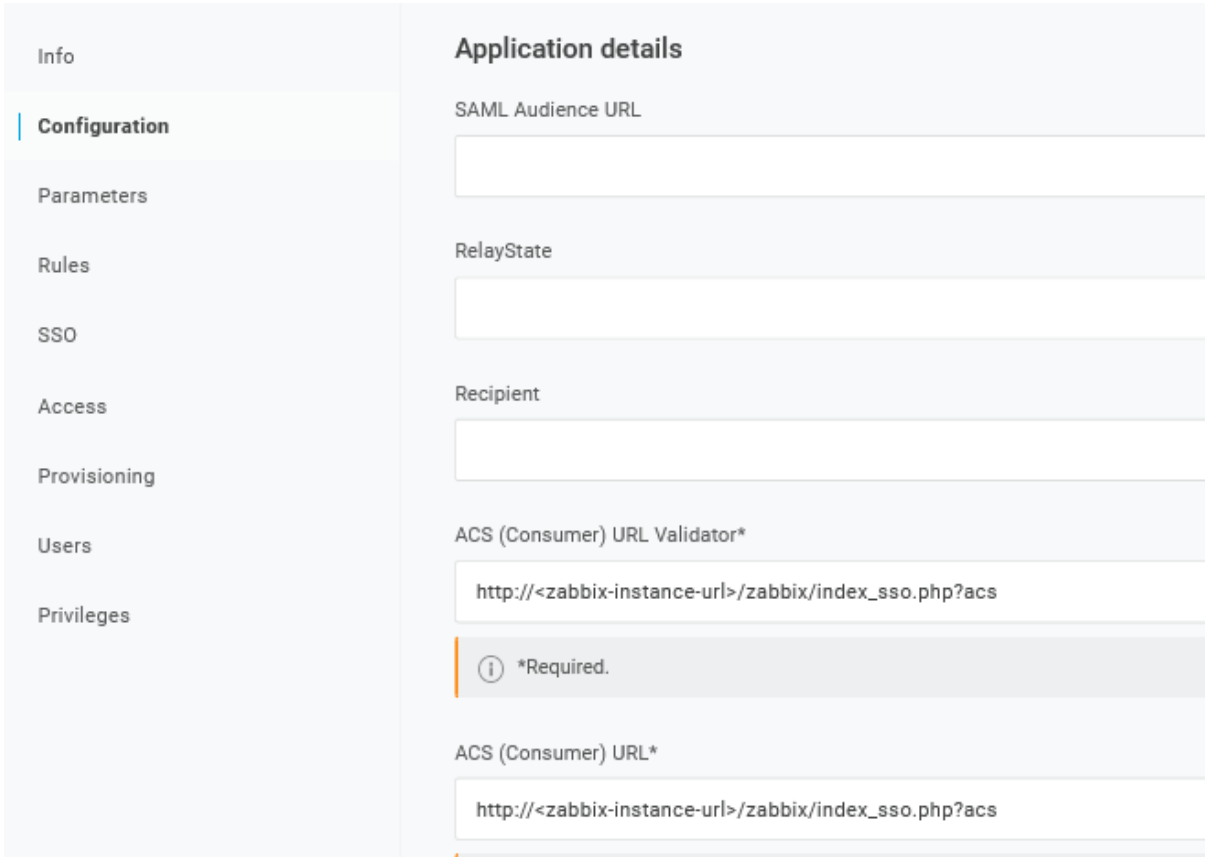
4. Na początek możesz chcieć dostosować nazwę wyświetlaną swojej aplikacji. Możesz także dodać ikonę i szczegóły aplikacji. Następnie kliknij *Save*.

Konfigurowanie aprowizacji SSO/SCIM

1. W sekcji *Configuration* > *Application details* ustaw punkt końcowy logowania jednokrotnego Zabbix `http://<zabbix-instance-url>/za` jako wartość następujących pól:

- *ACS (Consumer) URL Validator*
- *ACS (Consumer) URL*

Zwróć uwagę na użycie protokołu „http”, a nie „https”, aby parametr `acs` nie został usunięty z żądania.



Możliwe jest również użycie protokołu „https". Aby użyć go z Zabbix, dodaj następujący wiersz do pliku konfiguracyjnego frontendu (`zabbix.conf.php`):

```
$SSO['SETTINGS'] = ['use_proxy_headers' => true];
```

Pozostaw pozostałe opcje z ich wartościami domyślnymi.

2. W sekcji *Configuration* > *API connection* ustaw następujące wartości:

- *SCIM Base URL*: `https://<zabbix-instance-url>/zabbix/api_scim.php`
- *SCIM JSON Template*: powinien zawierać wszystkie atrybuty niestandardowe, które chcesz przekazywać do Zabbix za pośrednictwem SCIM, takie jak `user_name`, `user_lastname`, `user_email` i `user_mobile`:

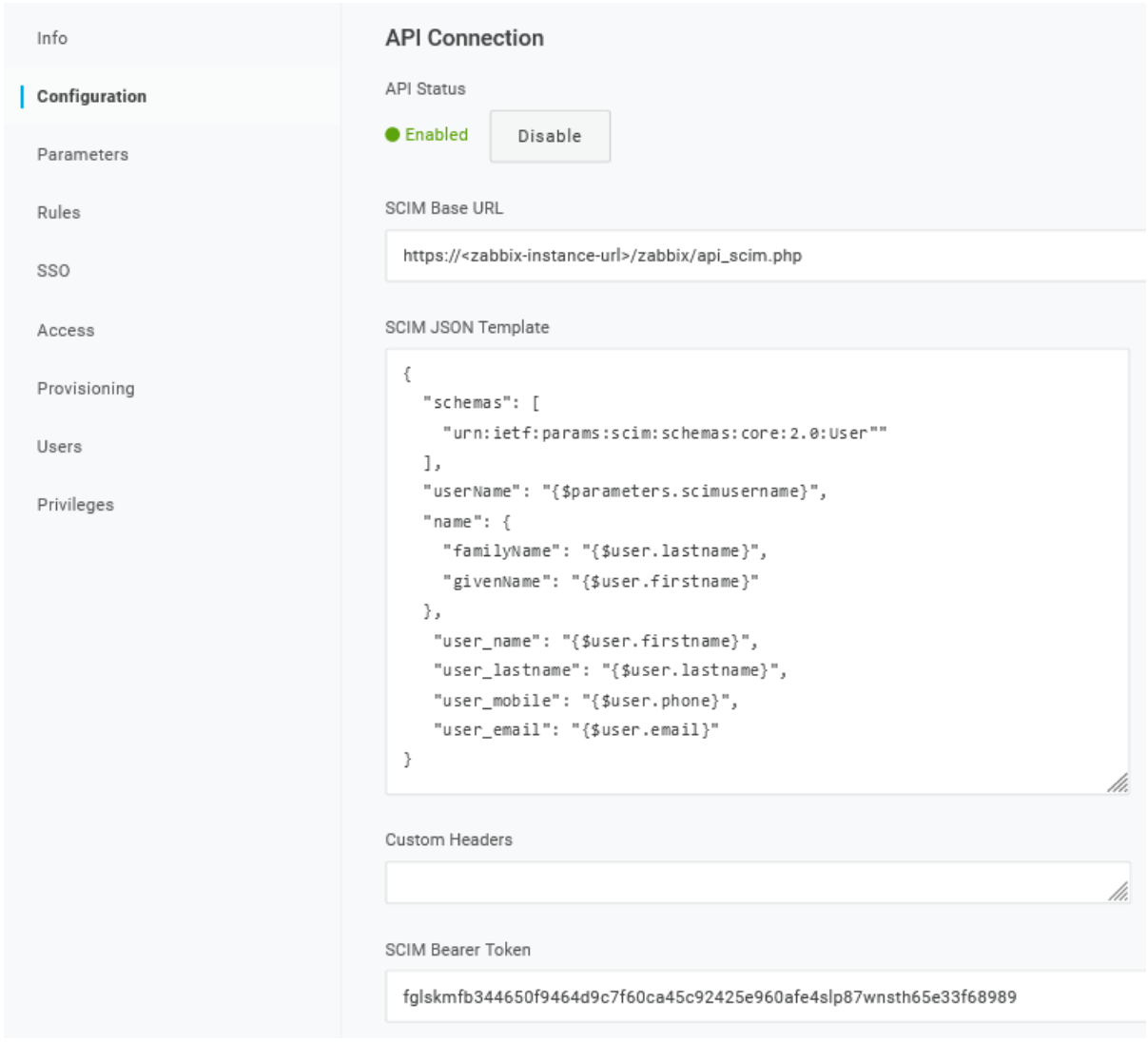
```
{
  "schemas": [
    "urn:ietf:params:scim:schemas:core:2.0:User"
  ],
  "userName": "{$parameters.scimusername}",
  "name": {
    "familyName": "{$user.lastname}",
    "givenName": "{$user.firstname}"
  },
   "user_name": "{$user.firstname}",
   "user_lastname": "{$user.lastname}",
   "user_mobile": "{$user.phone}",
   "user_email": "{$user.email}"
}
```

Nazwy atrybutów są dowolne. Można używać innych nazw atrybutów, jednak muszą one odpowiadać odpowiednim wartościom pól w ustawieniach SAML Zabbix.

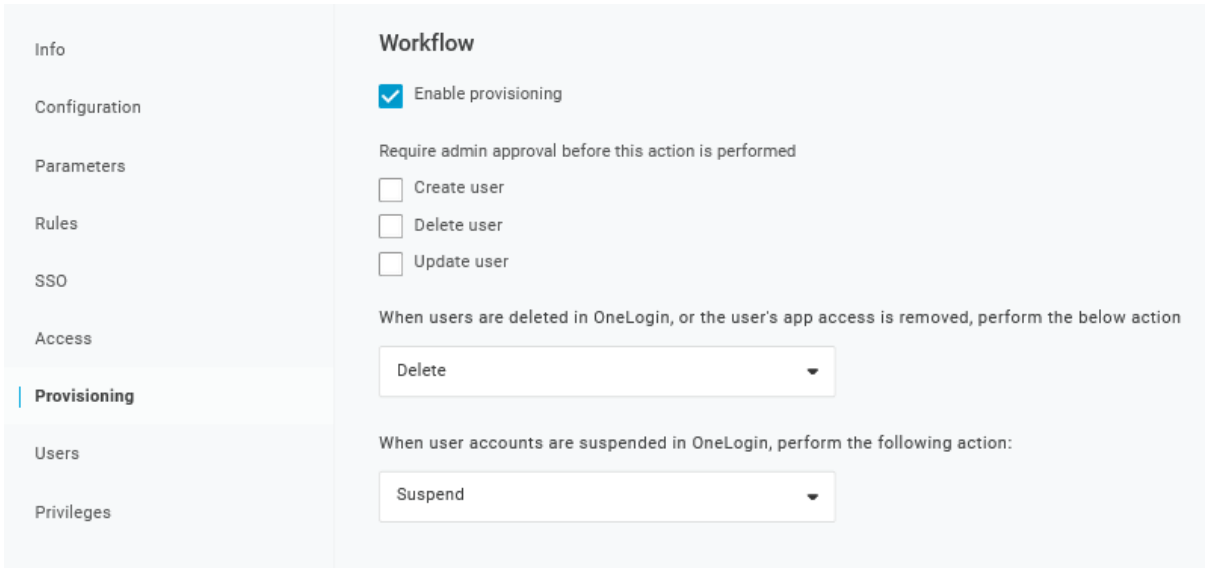
Pamiętaj, że aby aprowizacja użytkowników działała, OneLogin musi otrzymać w odpowiedzi atrybut „name" zawierający „givenName" i „familyName", nawet jeśli dostawca usług nie wymagał tych atrybutów. Dlatego należy określić je w schemacie w części konfiguracji aplikacji.

- *SCIM Bearer Token*: wprowadź token API Zabbix z uprawnieniami administratora Super admin.

Kliknij *Enable*, aby aktywować połączenie.

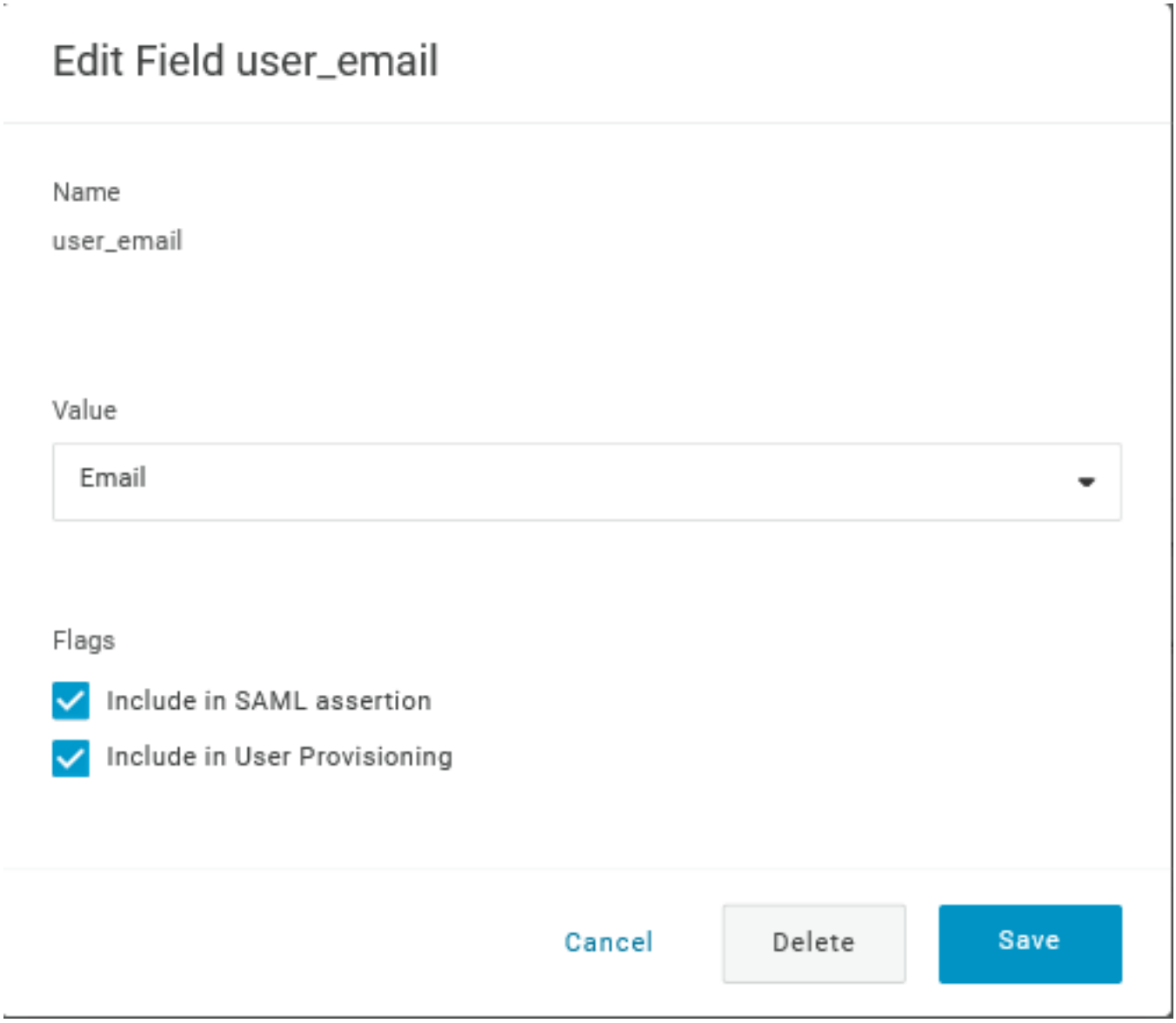


3. Na stronie *Provisioning* włącz opcję Provisioning:



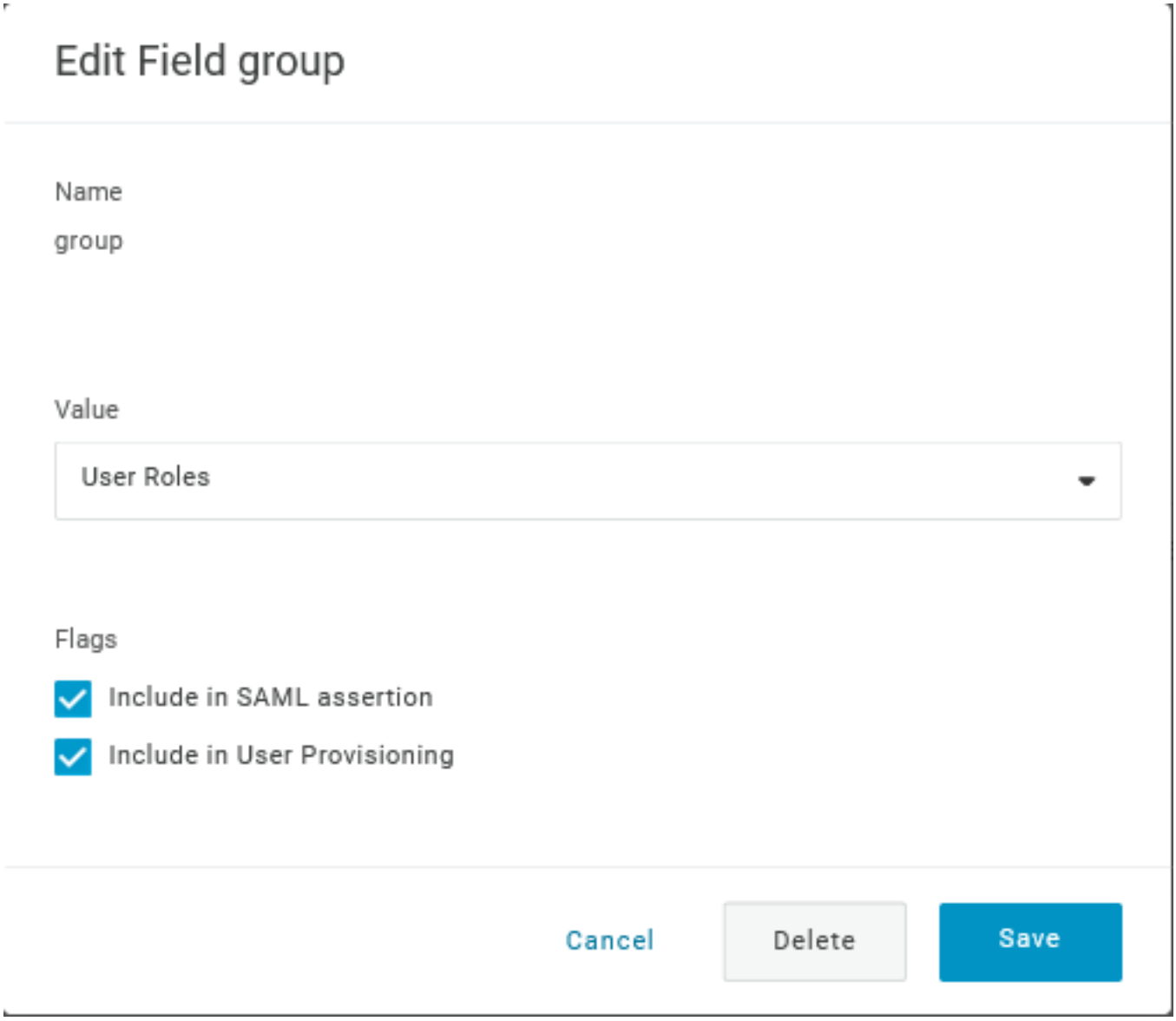
4. Strona *Parameters* zawiera listę parametrów domyślnych:

- Upewnij się, że parametr „scimusername” odpowiada wartości loginu użytkownika w OneLogin (np. adresowi e-mail);
- Zaznacz opcję *Include in User Provisioning* dla parametru „Groups”;
- Kliknij „+”, aby utworzyć parametry niestandardowe wymagane dla asercji SAML i aprowizacji użytkowników, takie jak `user_name`, `user_lastname`, `user_email` i `user_mobile`:

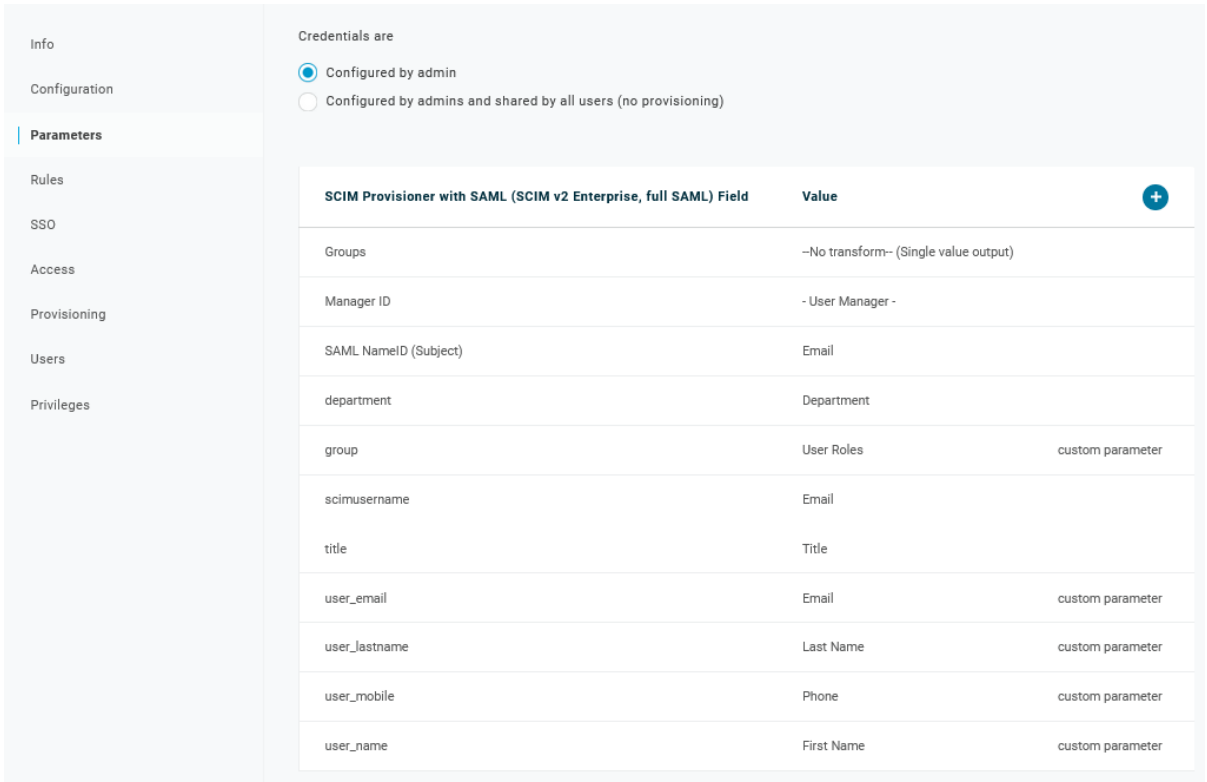


Podczas dodawania parametru zaznacz zarówno opcję *Include in SAML assertion*, jak i *Include in User Provisioning*.

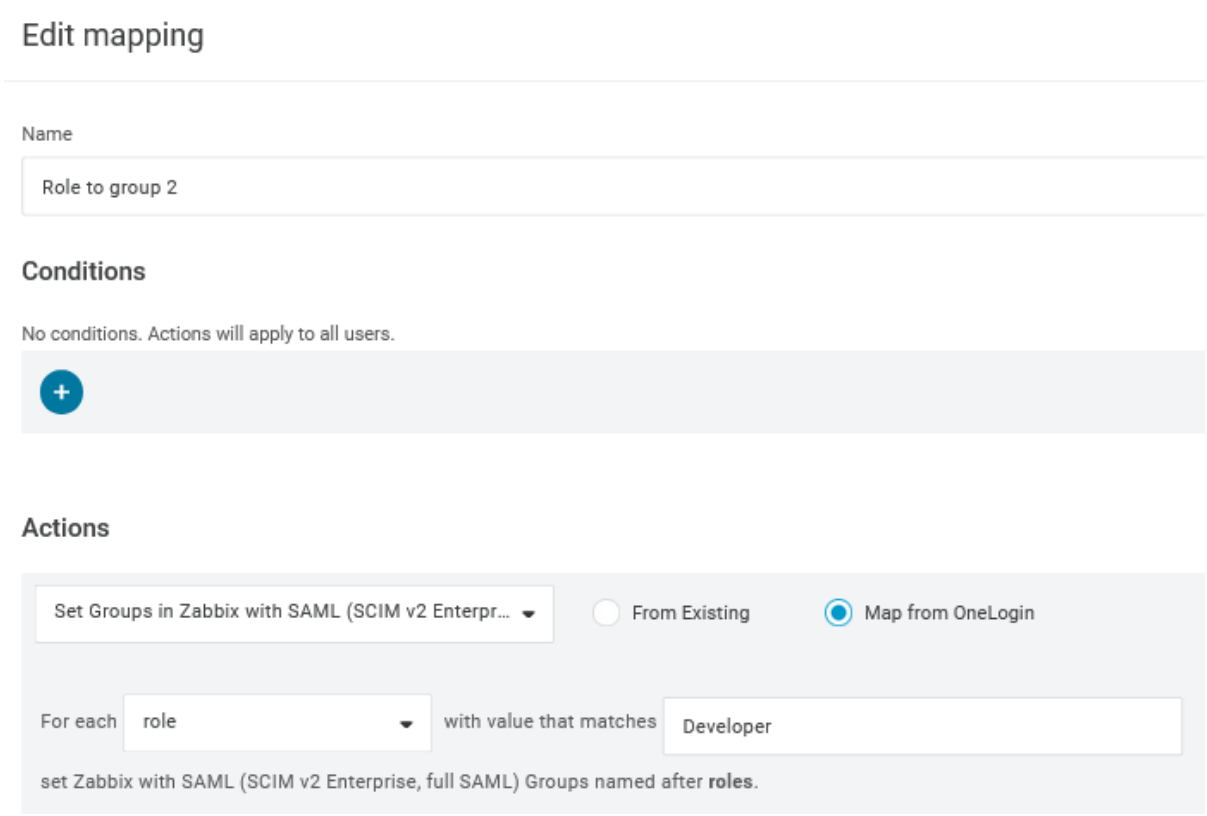
- Dodaj parametr „group” odpowiadający rolom użytkowników w OneLogin. Role użytkowników będą przekazywane jako ciąg znaków rozdzielonych średnikiem `;`. Role użytkowników OneLogin zostaną użyte do utworzenia grup użytkowników w Zabbix:



Zweryfikuj listę parametrów:



5. Na stronie *Rules* utwórz mapowania ról użytkowników do domyślnego parametru Groups.



Możesz użyć wyrażenia regularnego, aby przekazywać określone role jako grupy. Nazwy ról nie powinny zawierać `;`, ponieważ OneLogin używa tego znaku jako separatora podczas wysyłania atrybutu zawierającego kilka ról.

6. Pobierz certyfikat IdP. Ustaw dla niego uprawnienia 644, uruchamiając:

```
chmod 644 idp.crt
```

Konfiguracja Zabbix

1. W Zabbix przejdź do ustawień SAML i wypełnij opcje konfiguracji na podstawie konfiguracji OneLogin:



Pole Zabbix	Pole konfiguracji w OneLogin	Przykładowa wartość
IdP entity ID	Issuer URL (zobacz kartę SSO swojej aplikacji w OneLogin)	
SSO service URL	SAML 2.0 Endpoint (HTTP) (zobacz kartę SSO swojej aplikacji w OneLogin)	

Pole Zabbix	Pole konfiguracji w OneLogin	Przykładowa wartość
SLO service URL	SLO Endpoint (HTTP) (zobacz kartę SSO swojej aplikacji w 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`

Należy również skonfigurować mapowanie grup użytkowników. Mapowanie mediów jest opcjonalne. Kliknij *Update*, aby zapisać te ustawienia.

2. Dodaj certyfikat Base64 dostarczony przez OneLogin.

Jeśli w pliku konfiguracji frontendu (`zabbix.conf.php`) ustawiono `$SSO['CERT_STORAGE'] = 'database'`, podczas konfiguracji SAML możesz wkleić tekst certyfikatu lub przesłać plik certyfikatu w frontendzie — pliki w systemie plików nie są wymagane.

Jeśli w pliku konfiguracji ustawiono `$SSO['CERT_STORAGE'] = 'file'`, certyfikat musi być dostępny w systemie plików (domyślnie w `ui/conf/certs` lub w ścieżce skonfigurowanej w pliku konfiguracji), a frontend nie będzie przechowywać certyfikatów w bazie danych. Pamiętaj, że jeśli `$SSO['CERT_STORAGE']` nie jest ustawione lub jest zakomentowane, zakładane jest przechowywanie w pliku, a certyfikaty są odczytywane z `ui/conf/certs`.

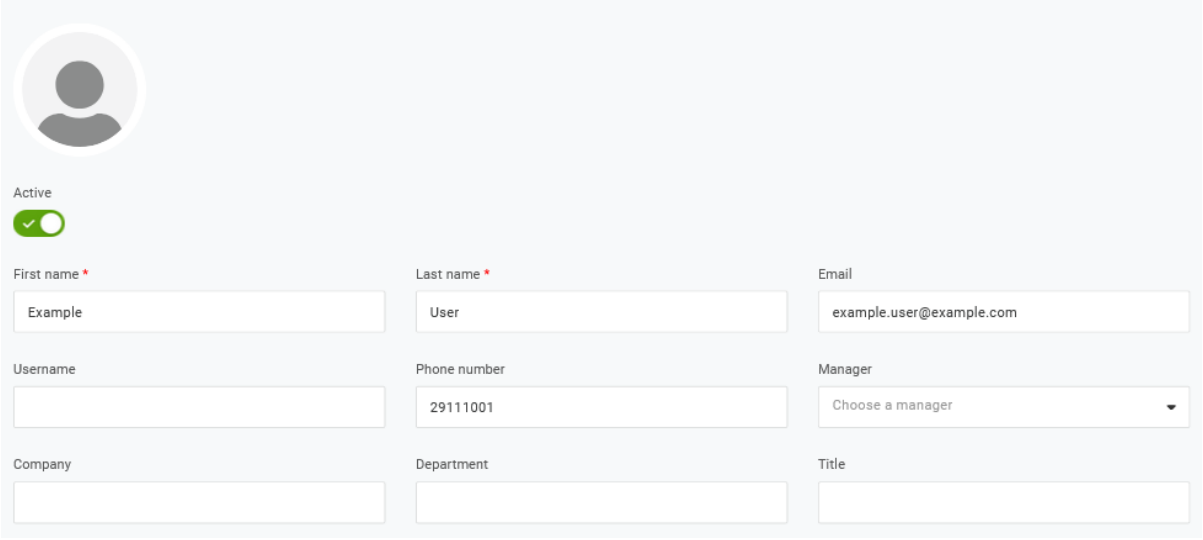
Dostęp do pobierania certyfikatu w OneLogin uzyskasz, wybierając *Applications* > *SSO* > klikając *View details* w sekcji bieżącego certyfikatu.

3. Kliknij przycisk *Update*, aby zapisać te ustawienia.

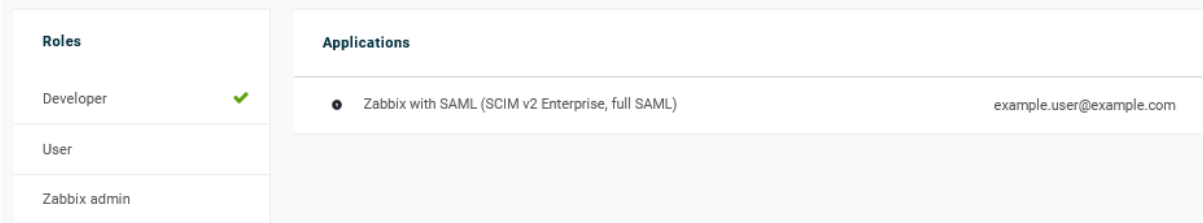
Provisioning użytkowników SCIM

Po włączeniu provisioning użytkowników można teraz dodawać/aktualizować użytkowników oraz ich role w OneLogin, a następnie natychmiast provisionować ich do Zabbix.

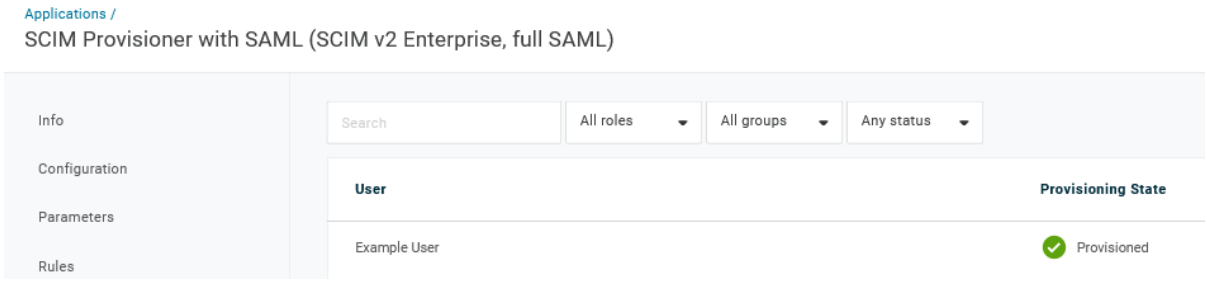
Na przykład można utworzyć nowego użytkownika:



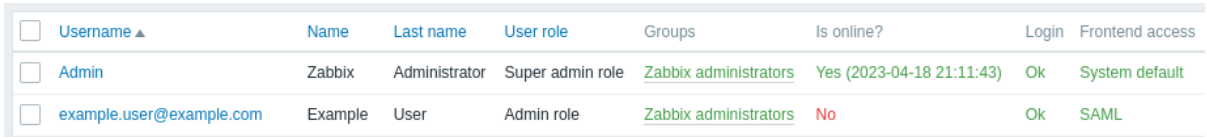
Dodaj go do roli użytkownika oraz aplikacji, która będzie provisionować użytkownika:



Po zapisaniu użytkownika zostanie on provisionowany do Zabbix. W *Application* > *Users* można sprawdzić status provisioning bieżących użytkowników aplikacji:



Jeśli provisioning zakończy się powodzeniem, użytkownik będzie widoczny na liście użytkowników 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

Konfigurowanie raportów cyklicznych

Przegląd

Ta sekcja zawiera instrukcje dotyczące instalacji usługi internetowej Zabbix oraz konfiguracji Zabbix w celu umożliwienia generowania zaplanowanych raportów.

Instalacja

Nowy proces Zabbix web service oraz przeglądarka Google Chrome powinny zostać zainstalowane, aby umożliwić generowanie zaplanowanych raportów. Usługa web service może zostać zainstalowana na tej samej maszynie, na której zainstalowany jest serwer Zabbix, lub na innej maszynie. Przeglądarka Google Chrome powinna być zainstalowana na tej samej maszynie, na której zainstalowana jest usługa web service.

Oficjalny pakiet zabbix-web-service jest dostępny w repozytorium Zabbix. Przeglądarka Google Chrome nie jest dołączona do tych pakietów i musi zostać zainstalowana osobno.

Aby skompilować usługę web Zabbix ze źródeł, zobacz Installing Zabbix web service.

Po instalacji uruchom `zabbix_web_service` na maszynie, na której zainstalowana jest usługa web:

```
zabbix_web_service
```

Konfiguracja

Aby zapewnić prawidłową komunikację między wszystkimi zaangażowanymi elementami, upewnij się, że parametry konfiguracji serwera i konfiguracji frontend są prawidłowo skonfigurowane.

Serwer Zabbix

Należy zaktualizować następujące parametry w pliku konfiguracyjnym serwera Zabbix: `WebServiceURL` i `StartReportWriters`.

WebServiceURL

Ten parametr jest wymagany do włączenia komunikacji z usługą sieciową. Adres URL powinien mieć format `http[s]://host:port/report`.

- Domyślnie usługa sieciowa nasłuchuje na porcie 10053. Inny port można określić w pliku konfiguracyjnym usługi sieciowej.
- Podanie ścieżki `/report` jest obowiązkowe (ścieżka jest zakodowana na stałe i nie można jej zmienić).

Przykład:

```
WebServiceURL=http://localhost:10053/report
```

Attention:
Zdecydowanie zaleca się skonfigurowanie szyfrowania między serwerem Zabbix a usługą sieciową Zabbix przy użyciu certyfikatów. Domyślnie dane przesyłane między serwerem Zabbix a usługą sieciową Zabbix nie są szyfrowane, co może prowadzić do nieautoryzowanego dostępu.

StartReportWriters

Ten parametr określa, ile procesów generowania raportów powinno zostać uruchomionych. Jeśli nie jest ustawiony lub ma wartość 0, generowanie raportów jest wyłączone. W zależności od liczby i częstotliwości wymaganych raportów można włączyć od 1 do 100 procesów generowania raportów.

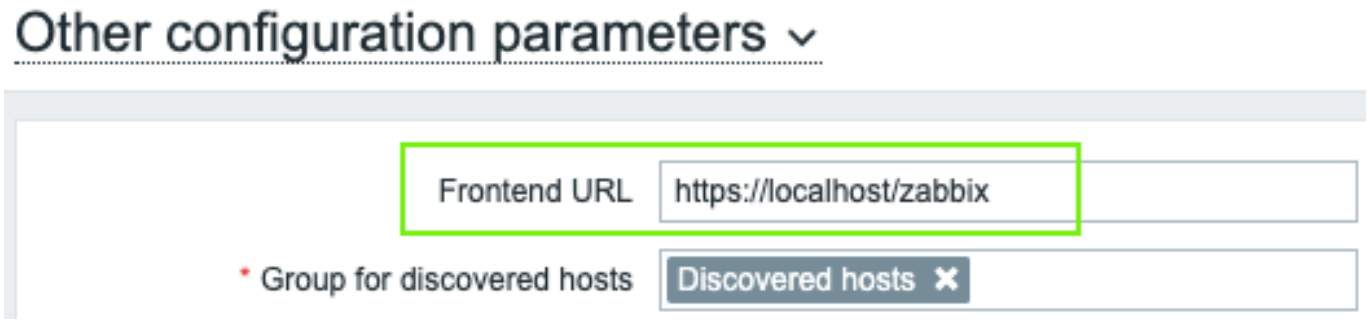
Przykład:

```
StartReportWriters=3
```

Zabbix frontend

Parametr *Frontend URL* powinien być ustawiony, aby umożliwić komunikację między Zabbix frontend a usługą internetową Zabbix:

- Przejdź do sekcji menu frontend *Administration* > *General* > *Other*
- Określ pełny adres URL interfejsu webowego Zabbix w parametrze *Frontend URL*



Note:
Po zakończeniu procedury konfiguracji możesz skonfigurować i wysłać raport testowy, aby upewnić się, że wszystko działa poprawnie.

Pozycje

Ta sekcja zawiera informacje referencyjne i uwagi techniczne dotyczące pozycji, w tym szczegóły parametrów dla określonych kluczy pozycji, informacje specyficzne dla platformy oraz uwagi dotyczące specjalnych konfiguracji pozycji.

Skorzystaj z paska bocznego, aby znaleźć informacje odpowiednie dla Twojego przypadku użycia.

Parametry vm.memory.size

Przegląd

Ta sekcja zawiera szczegółowe informacje o niektórych parametrach pozycji agent vm.memory.size[<mode>].

Parametry

Dla tej pozycji dostępne są następujące parametry:

- **active** - pamięć aktualnie używana lub używana jeszcze bardzo niedawno, a więc znajdująca się w RAM
- **anon** - pamięć niezwiązana z plikiem (nie można jej z niego ponownie odczytać)
- **available** - dostępna pamięć, obliczana różnie w zależności od platformy (zobacz tabelę poniżej)
- **buffers** - pamięć podręczna dla takich elementów jak metadane systemu plików
- **cached** - pamięć podręczna dla różnych danych
- **exec** - kod wykonywalny, zazwyczaj pochodzący z pliku (programu)
- **file** - pamięć podręczna zawartości ostatnio używanych plików
- **free** - pamięć, która jest od razu dostępna dla każdego podmiotu żądającego pamięci
- **inactive** - pamięć oznaczona jako nieużywana
- **pavailable** - pamięć „available” jako procent wartości „total” (obliczana jako `available/total*100`)
- **pinned** - to samo co „wired”
- **pused** - pamięć „used” jako procent wartości „total” (obliczana jako `used/total*100`)
- **shared** - pamięć, do której jednocześnie może uzyskiwać dostęp wiele procesów
- **slab** - całkowita ilość pamięci używanej przez jądro do buforowania struktur danych na własne potrzeby
- **total** - całkowita dostępna pamięć fizyczna
- **used** - używana pamięć, obliczana różnie w zależności od platformy (zobacz tabelę poniżej)
- **wired** - pamięć oznaczona jako mająca zawsze pozostawać w RAM. Nigdy nie jest przenoszona na dysk.

Warning:
Niektóre z tych parametrów są specyficzne dla platformy i mogą nie być dostępne na Twojej platformie. Szczegóły znajdują się w pozycjach agenta Zabbix.

Obliczanie wartości **available** i **used** zależnie od platformy:

Platforma	**"available"**	**"used"**
AIX	free + cached	rzeczywiście używana pamięć
FreeBSD	inactive + cached + free	active + wired + cached
HP UX	free	total - free
Linux<3.14	free + buffers + cached	total - free
Linux 3.14+ (również przeniesione do wersji 3.10 w RHEL 7)	/proc/meminfo, szczegóły znajdują się w sekcji „MemAvailable" dokumentacji jądra Linux documentation. Zwróć uwagę, że free + buffers + cached nie jest już równe „available", ponieważ nie cała pamięć podręczna stron może zostać zwolniona, a w obliczeniach używany jest niski próg znaku wodnego.	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:
Suma *vm.memory.size[used]* i *vm.memory.size[available]* nie musi być równa wartości total. Na przykład w FreeBSD:
 * Pamięć active, inactive, wired i cached jest uznawana za używaną, ponieważ przechowuje pewne użyteczne informacje.
 * Jednocześnie pamięć inactive, cached i free jest uznawana za dostępną, ponieważ te rodzaje pamięci mogą zostać natychmiast przydzielone procesom żądającym większej ilości pamięci.
Tak więc pamięć inactive jest jednocześnie używana i dostępna. Z tego powodu pozycja *vm.memory.size[used]* jest przeznaczona wyłącznie do celów informacyjnych, natomiast *vm.memory.size[available]* jest przeznaczona do użycia w wyzwalaczach.

Zobacz także

1. Dodatkowe szczegóły dotyczące obliczania pamięci w różnych Systemach Operacyjnych

Minimalny poziom uprawnień dla pozycji agenta Windows

Przegląd

Podczas monitorowania systemów przy użyciu agent, dobrą praktyką jest pozyskiwanie metryk z host, na którym agent jest zainstalowany. Aby stosować zasadę najmniejszych uprawnień, konieczne jest określenie, jakie metryki są pozyskiwane przez agent.

Tabela w tym dokumencie umożliwia wybór minimalnych uprawnień gwarantujących poprawne działanie Zabbix agent.

Jeśli do działania agent wybrano innego użytkownika niż „LocalSystem", to aby agent działał jako usługa systemu Windows, nowy użytkownik musi mieć uprawnienie „Log on as a service" w „Local Policy→User Rights Assignment" oraz prawo do tworzenia, zapisywania i usuwania pliku dziennika Zabbix agent. Użytkownik Active Directory musi zostać dodany do grupy *Performance Monitor Users*.

Note:
W przypadku pracy z uprawnieniami agent opartymi na grupie „minimalnie technicznie akceptowalnej" wymagane jest wcześniejsze nadanie uprawnień do obiektów przeznaczonych do monitorowania.

Obsługiwane standardowe pozycje agenta na Windows

Klucz pozycji	Grupa użytkowników	
	Zalecane	Minimalnie technicznie akceptowalne (funkcjonalność jest ograniczona)
agent.hostname	Goście	Goście

Klucz pozycji	Grupa użytkowników	
agent.ping	Goście	Goście
agent.variant	Goście	Goście
agent.version	Goście	Goście
log	Administratorzy	Goście
log.count	Administratorzy	Goście
logrt	Administratorzy	Goście
logrt.count	Administratorzy	Goście
net.dns	Goście	Goście
net.dns.perf	Goście	Goście
net.dns.record	Goście	Goście
net.if.discovery	Goście	Goście
net.if.in	Goście	Goście
net.if.out	Goście	Goście
net.if.total	Goście	Goście
net.tcp.listen	Goście	Goście
net.tcp.port	Goście	Goście
net.tcp.service	Goście	Goście
net.tcp.service.perf	Goście	Goście
net.udp.service	Goście	Goście
net.udp.service.perf	Goście	Goście
proc.num	Administratorzy	Goście
system.cpu.discovery	Użytkownicy Monitora wydajności	Użytkownicy Monitora wydajności
system.cpu.load	Użytkownicy Monitora wydajności	Użytkownicy Monitora wydajności
system.cpu.num	Goście	Goście
system.cpu.util	Użytkownicy Monitora wydajności	Użytkownicy Monitora wydajności
system.hostname	Goście	Goście
system.localtime	Goście	Goście
system.run	Administratorzy	Goście
system.sw.arch	Goście	Goście
system.swap.size	Goście	Goście
system.uname	Goście	Goście
system.uptime	Użytkownicy Monitora wydajności	Użytkownicy Monitora wydajności
vfs.dir.count	Administratorzy	Goście
vfs.dir.get	Administratorzy	Goście
vfs.dir.size	Administratorzy	Goście
vfs.file.cksum	Administratorzy	Goście
vfs.file.contents	Administratorzy	Goście
vfs.file.exists	Administratorzy	Goście
vfs.file.md5sum	Administratorzy	Goście
vfs.file.regexp	Administratorzy	Goście
vfs.file.regmatch	Administratorzy	Goście
vfs.file.size	Administratorzy	Goście
vfs.file.time	Administratorzy	Goście
vfs.fs.discovery	Administratorzy	Goście
vfs.fs.size	Administratorzy	Goście
vfs.fs.get	Administratorzy	Goście
vm.memory.size	Goście	Goście
web.page.get	Goście	Goście
web.page.perf	Goście	Goście
web.page.regexp	Goście	Goście
zabbix.stats	Goście	Goście

Klucze pozycji specyficzne dla systemu Windows

Item key	User group	
	Zalecane	Minimalne technicznie akceptowalne (funkcjonalność jest ograniczona)
eventlog	Event Log Readers	Guests
net.if.list	Guests	Guests
perf_counter	Performance Monitor Users	Performance Monitor Users

Item key	User group	
proc_info	Administrators	Guests
service.discovery	Guests	Guests
service.info	Guests	Guests
services	Guests	Guests
wmi.get	Administrators	Guests
vm.vmemory.size	Guests	Guests

Kodowanie zwracanych wartości

Serwer Zabbix oczekuje, że każda zwracana wartość tekstowa będzie zakodowana w UTF8. Dotyczy to każdego typu kontroli: Zabbix agent, SSH, Telnet itd.

Różne monitorowane systemy/urządzenia i kontrole mogą zwracać w wartości znaki spoza ASCII. W takich przypadkach prawie wszystkie możliwe klucze **zabbix** zawierają dodatkowy parametr klucza pozycji - **<encoding>**. Ten parametr klucza jest opcjonalny, ale należy go określić, jeśli zwracana wartość nie jest zakodowana w UTF8 i zawiera znaki spoza ASCII. W przeciwnym razie wynik może być nieoczekiwany i nieprzewidywalny.

Poniżej znajduje się opis zachowania w takich przypadkach dla różnych backendów bazy danych.

MySQL

Jeśli wartość zawiera znak spoza ASCII w kodowaniu innym niż UTF8, ten znak oraz kolejne zostaną odrzucone, gdy baza danych zapisze tę wartość. Do pliku `zabbix_server.log` nie zostaną zapisane żadne komunikaty ostrzegawcze.
 Dotyczy co najmniej wersji MySQL 5.1.61.

PostgreSQL

Jeśli wartość zawiera znak spoza ASCII w kodowaniu innym niż UTF8, spowoduje to niepowodzenie zapytania SQL (PGRES_FATAL_ERROR:ERROR invalid byte sequence for encoding) i dane nie zostaną zapisane. Odpowiedni komunikat ostrzegawczy zostanie zapisany w pliku `zabbix_server.log`.
 Dotyczy co najmniej wersji PostgreSQL 9.1.3.

Obsługa dużych plików

Obsługa dużych plików, często skracana do LFS, to termin odnoszący się do możliwości pracy z plikami większymi niż 2 GB w 32-bitowych systemach operacyjnych. Obsługa dużych plików wpływa co najmniej na monitorowanie plików dziennika oraz wszystkie pozycje vfs.file.*. Obsługa dużych plików zależy od możliwości systemu w czasie kompilacji Zabbix, ale jest całkowicie wyłączona w 32-bitowym Solarisie ze względu na niezgodność z procfs i swapctl.

Czujnik

Przegląd

Każdy układ czujnika ma własny katalog w drzewie sysfs `/sys/devices`. Aby znaleźć wszystkie układy czujników, łatwiej jest śledzić dowiązania symboliczne urządzeń z `/sys/class/hwmon/hwmon*`, gdzie `*` jest liczbą rzeczywistą (0,1,2,...).

Odczyty czujników znajdują się albo w katalogu `/sys/class/hwmon/hwmon*/` dla urządzeń wirtualnych, albo w katalogu `/sys/class/hwmon/hwmon*/device` dla urządzeń niewirtualnych. Plik o nazwie `name`, znajdujący się w katalogach `hwmon*` lub `hwmon*/device`, zawiera nazwę układu, która odpowiada nazwie sterownika jądra używanego przez układ czujnika.

W każdym pliku znajduje się tylko jedna wartość odczytu czujnika. Wspólny schemat nazewnictwa plików zawierających odczyty czujników wewnątrz dowolnego z wyżej wymienionych katalogów to: `<type><number>_<item>`, gdzie

- **type** - dla układów czujników jest to ”in” (napięcie), ”temp” (temperatura), ”fan” (wentylator) itd.
- **item** - ”input” (wartość zmierzona), ”max” (górny próg), ”min” (dolny próg) itd.
- **number** - zawsze używany dla elementów, które mogą występować więcej niż raz (zwykle zaczyna się od 1, z wyjątkiem napięć, które zaczynają się od 0). Jeśli pliki nie odnoszą się do konkretnego elementu, mają prostą nazwę bez numeru.

Informacje dotyczące czujników dostępnych na hoście można uzyskać za pomocą narzędzi **sensors-detect** i **sensors** (pakiet lm-sensors). **Sensors-detect** pomaga określić, które moduły są wymagane dla dostępnych czujników. Po załadowaniu modułów program **sensors** może zostać użyty do wyświetlenia odczytów wszystkich układów czujników. Etykiety odczytów czujników używane przez ten program mogą różnić się od wspólnego schematu nazewnictwa (`<type><number>_<item>`):

- jeśli istnieje plik o nazwie `<type><number>_label`, wtedy etykieta z tego pliku zostanie użyta zamiast nazwy `<type><number><item>`;

- jeśli nie ma pliku <type><number>_label, program przeszukuje /etc/sensors.conf (może to być także /etc/sensors3.conf lub inny plik) pod kątem podstawienia nazwy.

To etykietowanie pozwala użytkownikowi określić, jaki rodzaj sprzętu jest używany. Jeśli nie ma ani pliku <type><number>_label, ani etykiety w pliku konfiguracyjnym, typ sprzętu można określić na podstawie atrybutu name (hwmon*/device/name). Rzeczywiste nazwy czujników akceptowane przez zabbix_agent można uzyskać, uruchamiając program **sensors** z parametrem -u (sensors -u).

W programie **sensors** dostępne czujniki są rozdzielone według typu magistrali (adapter ISA, adapter PCI, adapter SPI, urządzenie wirtualne, interfejs ACPI, adapter HID).

Uzyskiwanie identyfikatorów sensorów

Etykiety sensorów, wyświetlane przez polecenie sensors, nie zawsze mogą być używane bezpośrednio, ponieważ sposób etykietowania różni się w zależności od producenta układu. Na przykład dane wyjściowe sensors mogą zawierać następujące wiersze:

```
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)
```

Spośród nich tylko jedna nazwa wyświetlana odpowiada bezpośrednio identyfikatorowi wewnętrznemu:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in0]
2.240000
```

Próba użycia innych etykiet wyświetlanych (takich jak Vcore lub +12V) nie zadziała:

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,Vcore]
ZBX_NOTSUPPORTED
```

Aby znaleźć wewnętrzny identyfikator sensora akceptowany przez Zabbix, uruchom sensors -u. W danych wyjściowych można zaobserwować następujące elementy:

```
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
...
```

Zatem etykieta Vcore odpowiada identyfikatorowi in1, a +12V odpowiada in4. Zgodnie ze specyfikacją kernel hwmon sysfs są to napięcia na pinach układu i ogólnie rzecz biorąc mogą wymagać skalowania.

```
zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in1]
1.301000
```

Zabbix może pobierać nie tylko odczyty napięcia (in), ale także natężenia prądu (curr), temperatury (temp) i prędkości wentylatora (fan).

Zgodność

Starsze dystrybucje

Odczyty czujników są pobierane z katalogu /proc/sys/dev/sensors:

- **device** - nazwa urządzenia (jeśli użyto <mode>, jest to wyrażenie regularne)
- **sensor** - nazwa czujnika (jeśli użyto <mode>, jest to wyrażenie regularne)
- **mode** - możliwe wartości: avg, max, min (jeśli ten parametr zostanie pominięty, device i sensor są traktowane dosłownie)

Przykładowy klucz: sensor[w83781d-i2c-0-2d,temp1].

Bieżące dystrybucje

Odczyty sensorów są pobierane z katalogu `/sys/class/hwmon`:

- **device** - nazwa urządzenia (bez wyrażeń regularnych). Nazwa urządzenia może być rzeczywistą nazwą urządzenia (np. `0000:00:18.3`) lub nazwą uzyskaną za pomocą programu sensors (np. `k8temp-pci-00c3`). Wybór, której nazwy użyć, należy do użytkownika.
- **sensor** - nazwa sensora (bez wyrażeń regularnych).
- **mode** - możliwe wartości: `avg`, `max`, `min` (jeśli ten parametr zostanie pominięty, device i sensor są traktowane dosłownie).

Przykładowe klucze:

`sensor[k8temp-pci-00c3,temp,max]` lub `sensor[0000:00:18.3,temp1]`.

`sensor[smsc47b397-isa-0880,in,avg]` lub `sensor[smsc47b397.2176,in1]`.

Uwagi dotyczące parametru memtype w pozycjach proc.mem

Omówienie

Parametr **memtype** jest obsługiwany na platformach Linux, AIX, FreeBSD i Solaris.

Na wszystkich tych platformach obsługiwane są trzy typowe wartości parametru `memtype`: `pmem`, `rss` i `vsize`. Ponadto na niektórych platformach obsługiwane są wartości `memtype` specyficzne dla danej platformy.

AIX

W tabeli przedstawiono wartości obsługiwane dla parametru `memtype` w systemie AIX.

Obsługiwana wartość	Opis	Źródło w strukturze procentry64	Próbuje zachować zgodność z
`vsize` [1]	Rozmiar pamięci wirtualnej	`pi_size`	
`pmem`	Procent pamięci rzeczywistej	`pi_prm`	`ps -o pmem`
`rss`	Rozmiar rezydentnego zbioru	`pi_trss + pi_drss`	`ps -o rssize`
`size`	Rozmiar procesu (kod + dane)	`pi_dvm`	Kolumna SIZE polecenia `ps gvw`
`dsize`	Rozmiar danych	`pi_dsize`	
`tsize`	Rozmiar tekstu (kodu)	`pi_tsize`	Kolumna TSIZ polecenia `ps gvw`
`sdsize`	Rozmiar danych z biblioteki współdzielonej	`pi_sdsize`	
`drss`	Rozmiar rezydentnego zbioru danych	`pi_drss`	
`trss`	Rozmiar rezydentnego zbioru tekstu	`pi_trss`	

Uwagi dotyczące AIX:

1. Podczas wybierania parametrów dla klucza pozycji `proc.mem[]` w systemie AIX spróbuj określić precyzyjne kryteria wyboru procesów. W przeciwnym razie istnieje ryzyko uwzględnienia niepożądanych procesów w wyniku `proc.mem[]`.

Przykład:

```
$ zabbix_agentd -t proc.mem[,,,NonExistingProcess,rss]
proc.mem[,,,NonExistingProcess,rss]           [u|2879488]
```

Ten przykład pokazuje, że określenie wyłącznie parametru wiersza poleceń (wyrażenia regularnego do dopasowania) powoduje uwzględnienie procesu agenta Zabbix, co prawdopodobnie nie jest oczekiwanym rezultatem.

2. Nie używaj `ps -ef` do przeglądania procesów — wyświetla ono tylko procesy inne niż jądra. Użyj `ps -Af`, aby zobaczyć wszystkie procesy widoczne dla agenta Zabbix.
3. Przeanalizujmy przykład `topasrec`, aby zobaczyć, jak agent Zabbix `proc.mem[]` wybiera procesy.

```
$ 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[]` ma następujące argumenty:

`proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]`

gdzie: - Pierwszym kryterium jest nazwa procesu (argument `<name>`). W naszym przykładzie agent Zabbix zobaczy ją jako `topasrec`. Aby uzyskać dopasowanie, należy określić `topasrec` lub pozostawić to pole puste. - Drugim kryterium jest nazwa użytkownika (argument `<user>`). Aby uzyskać dopasowanie, należy określić `root` lub pozostawić to pole puste. - Trzecim kryterium

używanym podczas wyboru procesu jest argument <cmdline>. agent Zabbix zobaczy jego wartość jako /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. Aby uzyskać dopasowanie, należy określić wyrażenie regularne pasujące do tego ciągu lub pozostawić to pole puste.

Argumenty <mode> i <memtype> są stosowane po użyciu trzech wymienionych powyżej kryteriów.

FreeBSD

W tabeli przedstawiono wartości obsługiwane dla parametru memtype w systemie FreeBSD.

Obsługiwana wartość	Opis	Źródło w strukturze kinfo_proc	Próbuje zapewnić zgodność z
vsize	Rozmiar pamięci wirtualnej	kp_eproc.e_vm.vm_map_size lub ki_size	ps -o vsz
pmem	Procent pamięci rzeczywistej	obliczany na podstawie rss	ps -o pmem
rss	Rozmiar rezydentnego zbioru stron	kp_eproc.e_vm.vm_rssize lub ki_rssize	ps -o rss
size [1]	Rozmiar procesu (kod + dane + stos)	tsize + dsize + ssize	
tsize	Rozmiar tekstu (kodu)	kp_eproc.e_vm.vm_tsize lub ki_tsize	ps -o tsiz
dsize	Rozmiar danych	kp_eproc.e_vm.vm_dsize lub ki_dsize	ps -o dsiz
ssize	Rozmiar stosu	kp_eproc.e_vm.vm_ssize lub ki_ssize	ps -o ssiz

Linux

Obsługiwane wartości parametru memtype w systemie Linux przedstawiono w tabeli.

Obsługiwana wartość	Opis	Źródło w pliku /proc/<pid>/status
vsize [1]	Rozmiar pamięci wirtualnej	VmSize
pmem	Procent pamięci rzeczywistej	(VmRSS/total_memory) * 100
rss	Rozmiar rezydentnego zbioru stron	VmRSS
data	Rozmiar segmentu danych	VmData
exe	Rozmiar segmentu kodu	VmExe
hwm	Szczytowy rozmiar rezydentnego zbioru stron	VmHWM
lck	Rozmiar zablokowanej pamięci	VmLck
lib	Rozmiar bibliotek współdzielonych	VmLib
peak	Szczytowy rozmiar pamięci wirtualnej	VmPeak
pin	Rozmiar przypiętych stron	VmPin
pte	Rozmiar wpisów tablicy stron	VmPTE
size	Rozmiar segmentów kodu + danych + stosu procesu	VmExe + VmData + VmStk
stk	Rozmiar segmentu stosu	VmStk
swap	Rozmiar używanej przestrzeni wymiany	VmSwap

Uwagi dotyczące systemu Linux:

1. Starsze jądra systemu Linux nie obsługują wszystkich wartości memtype. Na przykład jądra Linux 2.4 nie obsługują wartości hwm, pin, peak, pte i swap.
2. Zauważyliśmy, że samomonitorowanie procesu aktywnego sprawdzania agenta Zabbix za pomocą proc.mem[...,...,...,...,dat pokazuje wartość o 4 kB większą niż wartość zgłoszona w wierszu VmData pliku /proc/<pid>/status agenta. W momencie wykonywania pomiaru segment danych agenta zwiększa się o 4 kB, a następnie powraca do poprzedniego rozmiaru.

Solaris

W tabeli przedstawiono wartości obsługiwane dla parametru memtype w systemie Solaris.

Obsługiwana wartość	Opis	Źródło w strukturze psinfo	Próbuje zapewnić zgodność z
vsize [1]	Rozmiar obrazu procesu	pr_size	ps -o vsz
pmem	Procent rzeczywistej pamięci	pr_pctmem	ps -o pmem
rss	Rozmiar rezydentnego zbioru stron Może być zaniżony (zobacz opis rss w man ps).	pr_rssize	ps -o rss

Przypisy

[1] Wartość domyślna.

Uwagi dotyczące wybierania procesów w pozycjach proc.mem i proc.num

Procesy modyfikujące swoją linię poleceń

Niektóre programy wykorzystują modyfikowanie swojej linii poleceń jako metodę wyświetlania bieżącej aktywności. Użytkownik może zobaczyć tę aktywność, uruchamiając polecenia ps i top. Przykładami takich programów są *PostgreSQL*, *Sendmail*, *Zabbix*.

Zobaczmy przykład z Linuksa. Załóżmy, że chcemy monitorować liczbę procesów Zabbix agent.

Polecenie ps pokazuje interesujące nas procesy jako

```
$ 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]
...
```

Wybieranie procesów według nazwy i użytkownika działa:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd,zabbix]'
6
```

Teraz zmieńmy nazwę pliku wykonywalnego zabbix_agentd na zabbix_agentd_30 i uruchommy go ponownie.

ps pokazuje teraz

```
$ 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]
...
```

Teraz wybieranie procesów według nazwy i użytkownika daje niepoprawny wynik:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd_30,zabbix]'
1
```

Dlaczego zwykła zmiana nazwy pliku wykonywalnego na dłuższą prowadzi do tak odmiennego wyniku?

Zabbix agent zaczyna od sprawdzenia nazwy procesu. Otwierany jest plik /proc/<pid>/status i sprawdzany jest wiersz Name. W naszym przypadku wiersze Name są następujące:

```
$ 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
```

Nazwa procesu w pliku `status` jest obcinana do 15 znaków.

Podobny wynik można zobaczyć przy użyciu polecenia `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
 ...
```

Oczywiście nie jest to równe wartości parametru `name` `proc.num[]`, czyli `zabbix_agentd_30`. Po nieudanej próbie dopasowania nazwy procesu z pliku `status` Zabbix agent przechodzi do pliku `/proc/<pid>/cmdline`.

To, jak agent widzi plik „cmdline", można zilustrować, uruchamiając polecenie

```
$ 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
```

Pliki `/proc/<pid>/cmdline` w naszym przypadku zawierają niewidoczne, niedrukowalne bajty null, używane do kończenia łańcuchów w języku *C*. W tym przykładzie bajty null są pokazane jako „<NUL>".

Zabbix agent sprawdza „cmdline" dla procesu głównego i pobiera `zabbix_agentd_30`, co odpowiada wartości naszego parametru `name` `zabbix_agentd_30`. Dlatego proces główny jest zliczany przez pozycja `proc.num[zabbix_agentd_30,zabbix]`.

Podczas sprawdzania kolejnego procesu agent pobiera `zabbix_agentd_30: collector [idle 1 sec]` z pliku `cmdline` i nie spełnia to warunku naszego parametru `name` `zabbix_agentd_30`. Zatem zliczany jest tylko proces główny, który nie modyfikuje swojej linii poleceń. Pozostałe procesy agent modyfikują swoją linię poleceń i są ignorowane.

Ten przykład pokazuje, że parametr `name` nie może być używany w `proc.mem[]` i `proc.num[]` do wybierania procesów w takim przypadku.

Note:
Dla pozycja `proc.get[]`, gdy Zabbix agent sprawdza „cmdline" pod kątem nazwy procesu, użyje tylko tej części nazwy, która zaczyna się od ostatniego ukośnika i kończy na pierwszej spacji lub dwukropku. Nazwa procesu pobrana z pliku cmdline zostanie użyta tylko wtedy, gdy jej początek całkowicie odpowiada skróconej nazwie procesu w pliku `status`. Algorytm jest taki sam zarówno dla nazwy procesu w filtrze, jak i w wyjściu JSON.

Użycie parametru `cmdline` z odpowiednim wyrażeniem regularnym daje poprawny wynik:

```
$ zabbix_get -s localhost -k 'proc.num[,zabbix,,zabbix_agentd_30[ :]]'
6
```

Zachowaj ostrożność podczas używania pozycji `proc.get[]`, `proc.mem[]` i `proc.num[]` do monitorowania programów, które modyfikują swoje linie poleceń.

Przed wstawieniem parametrów `name` i `cmdline` do pozycji `proc.get[]`, `proc.mem[]` i `proc.num[]` możesz chcieć przetestować te parametry przy użyciu pozycji `proc.num[]` i polecenia `ps`.

Wątki jądra Linux

W pozycjach `proc.get[]`, `proc.mem[]` i `proc.num[]` nie można wybierać wątków za pomocą parametru `cmdline`

Jako przykład weźmy jeden z wątków jądra:

```
$ ps -ef| grep kthreadd
root         2     0  0 09:33 ?        00:00:00 [kthreadd]
```

Można go wybrać za pomocą parametru `name` procesu:

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd,root]'
1
```

Jednak wybieranie według parametru cmdline procesu nie działa:

```
$ zabbix_get -s localhost -k 'proc.num[,root,,kthreadd]'
0
```

Powodem jest to, że agent Zabbix pobiera wyrażenie regularne określone w parametrze cmdline i stosuje je do zawartości pliku procesu /proc/<pid>/cmdline. W przypadku wątków jądra ich pliki /proc/<pid>/cmdline są puste. Dlatego parametr cmdline nigdy nie pasuje.

Zliczanie wątków w pozycjach proc.mem[] i proc.num[]

Wątki jądra Linux są zliczane przez pozycję proc.num[], ale nie raportują pamięci w pozycji proc.mem[]. Na przykład:

```
$ 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.
```

Ale co się stanie, jeśli istnieje proces użytkownika o tej samej nazwie co wątek jądra? Wtedy może to wyglądać tak:

```
$ 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[] zliczył zarówno wątek jądra, jak i proces użytkownika. proc.mem[] raportuje pamięć tylko dla procesu użytkownika i traktuje pamięć wątku jądra tak, jakby wynosiła 0. Różni się to od przypadku powyżej, gdy zgłaszane było ZBX_NOTSUPPORTED.

Zachowaj ostrożność podczas używania pozycji proc.mem[] i proc.num[], jeśli nazwa programu przypadkowo odpowiada nazwie jednego z wątków.

Przed wstawieniem parametrów do pozycji proc.mem[] i proc.num[] warto przetestować parametry za pomocą pozycji proc.num[] oraz polecenia ps.

Szczegóły implementacji kontroli net.tcp.service i net.udp.service

Implementacja kontroli net.tcp.service i net.udp.service jest szczegółowo opisana na tej stronie dla różnych usług określonych w parametrze service.

Parametry pozycji net.tcp.service

ftp

Tworzy połączenie TCP i oczekuje, że pierwsze 4 znaki odpowiedzi będą równe "220 ", a następnie wysyła "QUIT\r\n". Jeśli nie określono portu, używany jest domyślny port 21.

http

Tworzy połączenie TCP bez oczekiwania i wysyłania czegokolwiek. Jeśli nie określono portu, używany jest domyślny port 80.

https

Używa biblioteki libcurl (i działa wyłącznie z nią), nie weryfikuje autentyczności certyfikatu, nie weryfikuje nazwy hosta w certyfikacie SSL, pobiera jedynie nagłówek odpowiedzi (żądanie HEAD). Jeśli nie określono portu, używany jest domyślny port 443.

imap

Tworzy połączenie TCP i oczekuje, że pierwsze 4 znaki odpowiedzi będą równe "* OK", a następnie wysyła "a1 LOGOUT\r\n". Jeśli nie określono portu, używany jest domyślny port 143.

ldap

Otwiera połączenie z serwerem LDAP i wykonuje operację wyszukiwania LDAP z filtrem ustawionym na (objectClass=*). Oczekuje pomyślnego pobrania pierwszego atrybutu pierwszego wpisu. Jeśli nie określono portu, używany jest domyślny port 389.

nntp

Tworzy połączenie TCP i oczekuje, że pierwsze 3 znaki odpowiedzi będą równe ”200” lub ”201”, a następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 119.

pop

Tworzy połączenie TCP i oczekuje, że pierwsze 3 znaki odpowiedzi będą równe ”+OK”, a następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 110.

smtp

Tworzy połączenie TCP i oczekuje, że pierwsze 3 znaki odpowiedzi będą równe ”220”, po których następuje spacja, znak końca linii lub myślnik. Linie zawierające myślnik należą do odpowiedzi wielowierszowej i odpowiedź będzie odczytywana ponownie aż do otrzymania linii bez myślnika. Następnie wysyła ”QUIT\r\n”. Jeśli nie określono portu, używany jest domyślny port 25.

ssh

Tworzy połączenie TCP. Jeśli połączenie zostało ustanowione, obie strony wymieniają ciąg identyfikacyjny (SSH-major.minor-XXXX), gdzie major i minor to wersje protokołu, a XXXX to ciąg znaków. Zabbix sprawdza, czy znaleziono ciąg zgodny ze specyfikacją, a następnie odsyła ciąg ”SSH-major.minor-zabbix_agent\r\n” lub ”0\n” w przypadku niezgodności. Jeśli nie określono portu, używany jest domyślny port 22.

tcp

Tworzy połączenie TCP bez oczekiwania i wysyłania czegokolwiek. W przeciwieństwie do innych kontroli wymaga określenia parametru portu.

telnet

Tworzy połączenie TCP i oczekuje monitu logowania (’:’ na końcu). Jeśli nie określono portu, używany jest domyślny port 23.

Parametry pozycji net.udp.service

ntp

Wysyła pakiet SNTP przez UDP i sprawdza poprawność odpowiedzi zgodnie z RFC 4330, sekcja 5. Jeśli nie określono portu, używany jest domyślny port 123.

Parametry proc.get

Przegląd

Pozycja **proc.get**[<name>,<user>,<cmdline>,<mode>] jest obsługiwana w systemach Linux, Windows, FreeBSD, OpenBSD i NetBSD.

Lista parametrów procesu zwracanych przez pozycję różni się w zależności od systemu operacyjnego i wartości argumentu „mode”.

Linux

W systemie Linux dla każdego trybu zwracane są następujące parametry procesu:

mode=process	mode=thread	mode=summary
pid: PID	pid: PID	name: nazwa procesu
ppid: nadrzędny PID	ppid: nadrzędny PID	processes: liczba procesów
name: nazwa procesu	name: nazwa procesu	vsize: rozmiar pamięci wirtualnej
cmdline: polecenie z argumentami	user: użytkownik (rzeczywisty), w którego kontekście działa proces	pmem: procent pamięci rzeczywistej
user: użytkownik (rzeczywisty), w którego kontekście działa proces	group: grupa (rzeczywista), w której kontekście działa proces	rss: rozmiar rezydentnego zbioru stron
group: grupa (rzeczywista), w której kontekście działa proces	uid: identyfikator użytkownika	data: rozmiar segmentu danych
uid: identyfikator użytkownika	gid: identyfikator grupy, w której kontekście działa proces	exe: rozmiar segmentu kodu
gid: identyfikator grupy, w której kontekście działa proces	tid: identyfikator wątku	lib: rozmiar bibliotek współdzielonych
vsize: rozmiar pamięci wirtualnej	tname: nazwa wątku	lck: rozmiar zablokowanej pamięci

mode=process	mode=thread	mode=summary
pmem: procent pamięci rzeczywistej	cputime_user: całkowity czas procesora spędzony w trybie użytkownika (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)	pin: rozmiar przypiętych stron
rss: rozmiar rezydentnego zbioru stron	cputime_system: całkowity czas procesora spędzony w trybie systemowym (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)	pte: rozmiar wpisów tablicy stron
data: rozmiar segmentu danych	state: stan wątku	size: rozmiar segmentów kodu + danych + stosu procesu
exe: rozmiar segmentu kodu	ctx_switches: liczba przełączeń kontekstu	stk: rozmiar segmentu stosu
hwm: szczytowy rozmiar rezydentnego zbioru stron	page_faults: liczba błędów stron	swap: rozmiar użytej przestrzeni wymiany
lck: rozmiar zablokowanej pamięci		cputime_user: całkowity czas procesora spędzony w trybie użytkownika (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)
lib: rozmiar bibliotek współdzielonych		cputime_system: całkowity czas procesora spędzony w trybie systemowym (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)
peak: szczytowy rozmiar pamięci wirtualnej		ctx_switches: liczba przełączeń kontekstu
pin: rozmiar przypiętych stron		threads: liczba wątków
pte: rozmiar wpisów tablicy stron		page_faults: liczba błędów stron
size: rozmiar segmentów kodu + danych + stosu procesu		pss: proporcjonalny rozmiar zbioru stron pamięci
stk: rozmiar segmentu stosu		
swap: rozmiar użytej przestrzeni wymiany		
cputime_user: całkowity czas procesora spędzony w trybie użytkownika (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)		
cputime_system: całkowity czas procesora spędzony w trybie systemowym (wartość wyrażona w taktach zegara — podziel przez `getconf CLK_TCK` lub `sysconf(_SC_CLK_TCK)`, aby uzyskać sekundy)		
state: stan procesu (pobierany bezpośrednio z procfs w pełnej postaci)		
ctx_switches: liczba przełączeń kontekstu		

mode=process	mode=thread	mode=summary
threads: liczba wątków page_faults: liczba błędów stron pss: proporcjonalny rozmiar zbioru stron pamięci		

Systemy operacyjne oparte na BSD

Następujące parametry procesu są zwracane w systemach FreeBSD, OpenBSD i NetBSD dla każdego trybu:

mode=process	mode=thread	mode=summary
pid: PID	pid: PID	name: nazwa procesu
ppid: PID procesu nadrzędnego	ppid: PID procesu nadrzędnego	processes: liczba procesów
jid: ID jaila (tylko FreeBSD)	jid: ID jaila (tylko FreeBSD)	vsize: rozmiar pamięci wirtualnej
jname: nazwa jaila (tylko FreeBSD)	jname: nazwa jaila (tylko FreeBSD)	pmem: procent pamięci rzeczywistej (tylko FreeBSD)
name: nazwa procesu	name: nazwa procesu	rss: rozmiar rezydentnego zestawu pamięci
cmdline: polecenie z argumentami	user: użytkownik (rzeczywisty), z którego uprawnieniami działa proces	size: rozmiar procesu (kod + dane + stos)
user: użytkownik (rzeczywisty), z którego uprawnieniami działa proces	group: grupa (rzeczywista), z której uprawnieniami działa proces	tsize: rozmiar tekstu (kodu)
group: grupa (rzeczywista), z której uprawnieniami działa proces	uid: ID użytkownika	dsize: rozmiar danych
uid: ID użytkownika	gid: ID grupy, z której uprawnieniami działa proces	ssize: rozmiar stosu
gid: ID grupy, z której uprawnieniami działa proces	tid: ID wątku	cputime_user: całkowity czas CPU spędzony w trybie użytkownika (podawany w sekundach)
vsize: rozmiar pamięci wirtualnej	tname: nazwa wątku	cputime_system: całkowity czas CPU spędzony w trybie systemowym (podawany w sekundach)
pmem: procent pamięci rzeczywistej (tylko FreeBSD)	cputime_user: całkowity czas CPU spędzony w trybie użytkownika (podawany w sekundach)	ctx_switches: liczba przełączeń kontekstu
rss: rozmiar rezydentnego zestawu pamięci	cputime_system: całkowity czas CPU spędzony w trybie systemowym (podawany w sekundach)	threads: liczba wątków (nieobsługiwane w NetBSD)
size: rozmiar procesu (kod + dane + stos)	state: stan wątku	stk: rozmiar segmentu stosu
tsize: rozmiar tekstu (kodu)	ctx_switches: liczba przełączeń kontekstu	page_faults: liczba błędów strony
dsize: rozmiar danych	io_read_op: liczba operacji wejścia wymaganych przez system	fds: liczba deskryptorów plików (tylko OpenBSD)
ssize: rozmiar stosu	io_write_op: liczba operacji wyjścia wymaganych przez system	swap: rozmiar używanej przestrzeni wymiany
cputime_user: całkowity czas CPU spędzony w trybie użytkownika (podawany w sekundach)		io_read_op: liczba operacji wejścia wymaganych przez system
cputime_system: całkowity czas CPU spędzony w trybie systemowym (podawany w sekundach)		io_write_op: liczba operacji wyjścia wymaganych przez system
state: stan procesu (uśpienie dyskowe/działający/uśpiony/zatrzymanie śledzenia/zombie/inny)		
ctx_switches: liczba przełączeń kontekstu		
threads: liczba wątków (nieobsługiwane w NetBSD)		
page_faults: liczba błędów strony		

mode=process	mode=thread	mode=summary
fds: liczba deskryptorów plików (tylko OpenBSD) swap: rozmiar używanej przestrzeni wymiany io_read_op: liczba operacji wejścia wymaganych przez system io_write_op: liczba operacji wyjścia wymaganych przez system		

Windows

Następujące parametry procesu są zwracane w systemie Windows dla każdego trybu:

mode=process	mode=thread	mode=summary
pid: PID ppid: PID procesu nadrzędnego name: nazwa procesu user: użytkownik, z uprawnieniami którego działa proces sid: SID użytkownika vmsize: rozmiar pamięci wirtualnej wkset: rozmiar zestawu roboczego procesu cputime_user: całkowity czas CPU spędzony w trybie użytkownika (raportowany w milisekundach) cputime_system: całkowity czas CPU spędzony w trybie systemowym (raportowany w milisekundach) threads: liczba wątków page_faults: liczba błędów strony handles: liczba uchwytów io_read_b: odczytane bajty IO io_write_b: zapisane bajty IO io_read_op: operacje odczytu IO io_write_op: operacje zapisu IO io_other_b: bajty IO przesłane w operacjach innych niż odczyt i zapis io_other_op: operacje IO inne niż odczyt i zapis	pid: PID ppid: PID procesu nadrzędnego name: nazwa procesu user: użytkownik, z uprawnieniami którego działa proces sid: SID użytkownika tid: ID wątku	name: nazwa procesu processes: liczba procesów vmsize: rozmiar pamięci wirtualnej wkset: rozmiar zestawu roboczego procesu cputime_user: całkowity czas CPU spędzony w trybie użytkownika (raportowany w milisekundach) cputime_system: całkowity czas CPU spędzony w trybie systemowym (raportowany w milisekundach) threads: liczba wątków page_faults: liczba błędów strony handles: liczba uchwytów io_read_b: odczytane bajty IO io_write_b: zapisane bajty IO io_read_op: operacje odczytu IO io_write_op: operacje zapisu IO io_other_b: bajty IO przesłane w operacjach innych niż odczyt i zapis io_other_op: operacje IO inne niż odczyt i zapis

Ustawienia niedostępnego/niewidocznego interfejsu hosta

Przegląd

Kilka parametrów konfiguracji parameters określa, jak serwer Zabbix powinien się zachować, gdy sprawdzenie agentem (Zabbix, SNMP, IPMI, JMX) zakończy się niepowodzeniem, a interfejs hosta stanie się niedostępny.

Niedostępny interfejs

Interfejs hosta jest uznawany przez Zabbix za niedostępny po nieudanym sprawdzeniu (błąd sieci, przekroczenie limitu czasu) przez agentów Zabbix, SNMP, IPMI lub JMX. Od Zabbix 6.2.0 aktywne sprawdzenia agenta Zabbix również wpływają na dostępność interfejsu. Gdy aktywne sprawdzenia stają się niedostępne, wpływają na ogólny status dostępności interfejsu agenta.

Od momentu, gdy interfejs staje się niedostępny, **UnreachableDelay** określa, jak często jest on ponownie sprawdzany przy użyciu jednej z pozycji (w tym reguł LLD). Te ponowne sprawdzenia są wykonywane przez pollery dla niedostępnych interfejsów (lub pollery IPMI dla sprawdzeń IPMI). Domyślnie odstęp między kolejnymi sprawdzeniami dostępności wynosi 15 sekund.

Attention:
Sprawdzenia wykonywane przez asynchroniczne pollery nie są przenoszone do pollerów dla niedostępnych interfejsów.

W logu serwera Zabbix niedostępność jest sygnalizowana komunikatami takimi jak:

```
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
```

Komunikaty logu określają dokładną pozycję, która zakończyła się niepowodzeniem, oraz jej typ (Zabbix agent).

Note:
Parametr *Timeout* wpływa również na to, jak wcześnie interfejs jest ponownie sprawdzany podczas niedostępności. Jeśli Timeout jest ustawiony na 20 sekund, a UnreachableDelay na 30 sekund, następne sprawdzenie nastąpi po 50 sekundach od pierwszej próby.

Parametr **UnreachablePeriod** określa całkowity czas trwania okresu niedostępności. Domyślnie UnreachablePeriod wynosi 45 sekund. Ta wartość powinna być kilkukrotnie większa niż UnreachableDelay, aby zapewnić, że interfejs zostanie sprawdzony ponownie wielokrotnie, zanim zostanie oznaczony jako niedostępny.

Pozycja wewnętrzna, zabbix[host,active_agent,available], umożliwia monitorowanie dostępności aktywnych sprawdzeń w scenariuszach niedostępności.

Przełączanie interfejsu z powrotem na dostępny

Po zakończeniu okresu niedostępności interfejs jest ponownie odpytywany, ze zmniejszonym priorytetem dla pozycji, która spowodowała przejście interfejsu w stan niedostępności. Jeśli niedostępny interfejs pojawi się ponownie, monitorowanie automatycznie wraca do normy:

```
wznowienie kontroli Zabbix agent na hoście "New host": połączenie przywrócone
```

Note:
Gdy interfejs staje się dostępny, host nie odpytuje natychmiast wszystkich swoich pozycji z dwóch powodów:
- Mogłoby to przeciążyć host.
- Czas przywrócenia interfejsu nie zawsze pokrywa się z planowanym czasem harmonogramu odpytywania pozycji.

Dlatego po przywróceniu dostępności interfejsu pozycje nie są odpytywane natychmiast, lecz są przeplanowywane na następną rundę odpytywania.

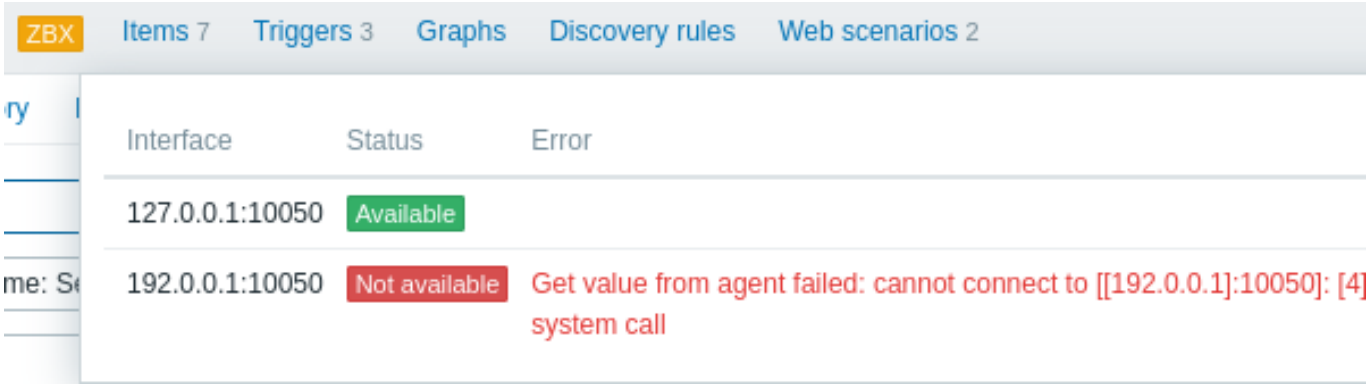
Niedostępny interfejs

Po zakończeniu okresu UnreachablePeriod, jeśli interfejs nie pojawi się ponownie, jest on traktowany jako niedostępny.

W logu serwera jest to sygnalizowane komunikatami takimi jak:

```
temporarily disabling Zabbix agent checks on host "New host": interface unavailable
```

a we frontend ikona dostępności hosta zmienia kolor z zielonego/szarego na żółty/czerwony (szczegóły niedostępnego interfejsu można zobaczyć w dymku podpowiedzi, który jest wyświetlany po najechaniu kursorem na ikonę dostępności hosta):



Parametr **UnavailableDelay** określa, jak często interfejs jest sprawdzany w czasie jego niedostępności.

Domyślnie jest to 60 sekund (więc w tym przypadku „temporarily disabling” z komunikatu logu powyżej będzie oznaczać wyłączenie sprawdzania na jedną minutę).

Gdy połączenie z interfejsem zostanie przywrócone, monitorowanie również automatycznie wraca do normy:

```
enabling Zabbix agent checks on host "New host": interface became available
```

Zdalne monitorowanie statystyk Zabbix

Przegląd

Możliwe jest udostępnienie zdalnie niektórych wewnętrznych metryk serwera Zabbix i proxy innemu egzemplarzowi Zabbix lub narzędziu firm trzecich. Może to być przydatne, aby dostawcy wsparcia/usług mogli monitorować zdalnie serwery/proxy Zabbix swoich klientów lub, w organizacjach, w których Zabbix nie jest głównym narzędziem monitorującym, aby wewnętrzne metryki Zabbix mogły być monitorowane przez system firm trzecich w konfiguracji nadrzędnego monitoringu.

Wewnętrzne statystyki Zabbix są udostępniane dla konfigurowalnego zestawu adresów wymienionych w nowym parametrze 'StatsAllowedIP' server/proxy. Żądania będą akceptowane tylko z tych adresów.

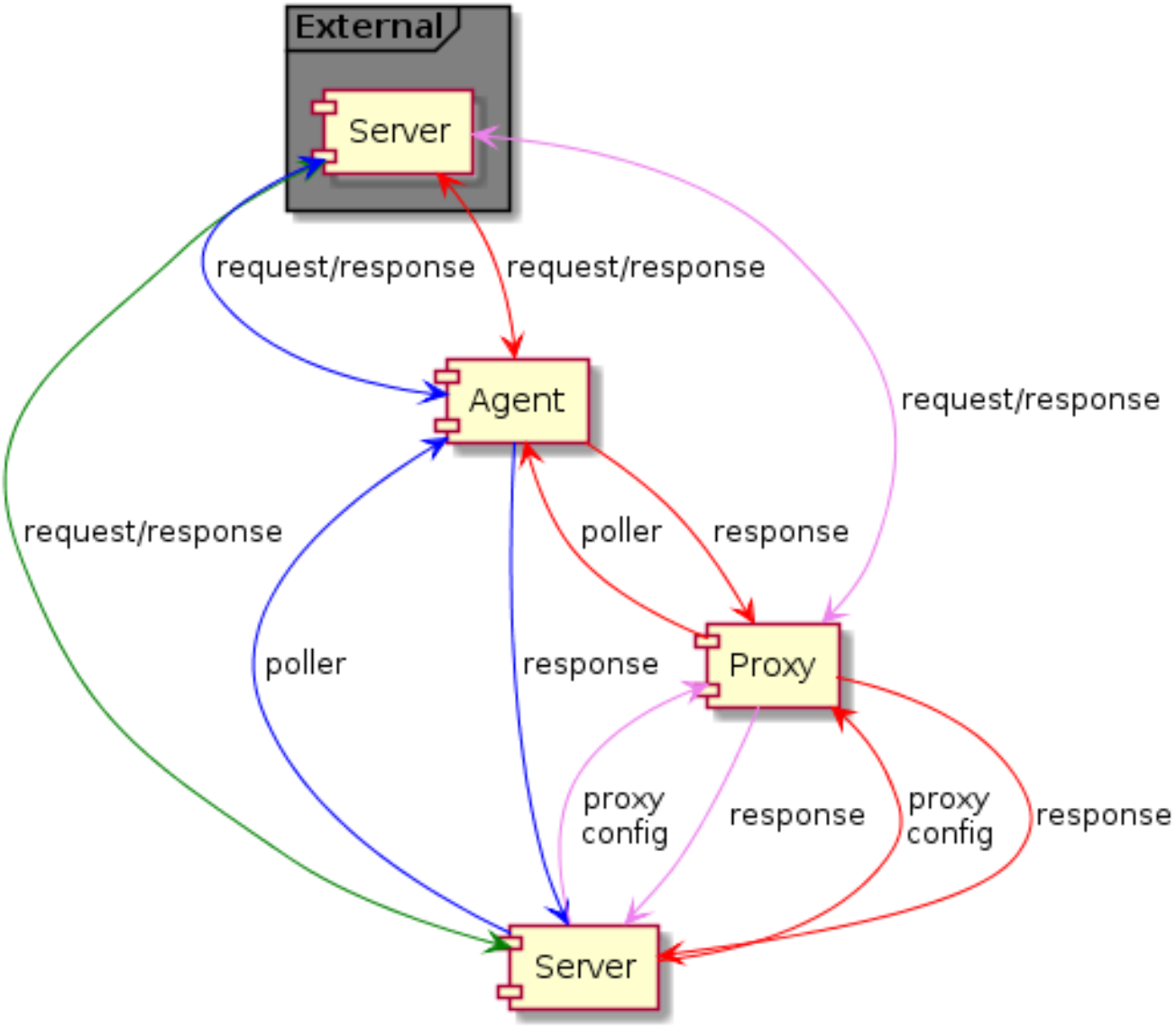
Pozycje

Aby skonfigurować odpytywanie wewnętrznych statystyk na innej instancji Zabbix, można użyć dwóch pozycji:

- wewnętrzna pozycja `zabbix[stats,<ip>,<port>]` — do bezpośrednich zdalnych zapytań do serwera/proxy Zabbix. `<ip>` i `<port>` są używane do identyfikacji instancji docelowej.
- pozycja agenta `zabbix.stats[<ip>,<port>]` — do zdalnych zapytań do serwera/proxy Zabbix opartych na agencie. `<ip>` i `<port>` są używane do identyfikacji instancji docelowej.

Zobacz także: Pozycje wewnętrzne, Pozycje agenta Zabbix

Poniższy diagram ilustruje użycie jednej z tych pozycji w zależności od kontekstu.



- — Serwer → zewnętrzna instancja Zabbix (`zabbix[stats,<ip>,<port>]`)
- — Serwer → proxy → zewnętrzna instancja Zabbix (`zabbix[stats,<ip>,<port>]`)
- — Serwer → agent → zewnętrzna instancja Zabbix (`zabbix.stats[<ip>,<port>]`)
- — Serwer → proxy → agent → zewnętrzna instancja Zabbix (`zabbix.stats[<ip>,<port>]`)

Aby upewnić się, że instancja docelowa zezwala na odpytywanie jej przez instancję zewnętrzną, podaj adres instancji zewnętrznej w parametrze `StatsAllowedIP` na instancji docelowej.

Udostępniane metryki

Pozycje statystyk zbierają statystyki zbiorczo i zwracają JSON, który stanowi podstawę dla pozycji zależnych do pobierania z niego danych. Następujące metryki wewnętrzne są zwracane przez dowolną z tych dwóch pozycji:

- `zabbix[boottime]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[preprocessing]` (tylko serwer)
- `zabbix[preprocessing_queue]` (tylko serwer)
- `zabbix[process,<type>,<mode>,<state>]` (tylko statystyki oparte na typie procesu)
- `zabbix[rcache,<cache>,<mode>]`
- `zabbix[requiredperformance]`
- `zabbix[triggers]` (tylko serwer)
- `zabbix[uptime]`
- `zabbix[vcache,buffer,<mode>]` (tylko serwer)

- zabbix[vcache,cache,<parameter>]
- zabbix[version]
- zabbix[vmware,buffer,<mode>]
- zabbix[wcache,<cache>,<mode>] (typ pamięci podręcznej „trends” tylko dla serwera)

Szablony

Szablony są dostępne do zdalnego monitorowania wewnętrznych metryk serwera Zabbix lub proxy z zewnętrznej instancji:

- Stan zdalnego serwera Zabbix
- Stan zdalnego proxy Zabbix

Należy pamiętać, że aby użyć szablonu do zdalnego monitorowania wielu zewnętrznych instancji, wymagany jest oddzielny host dla monitorowania każdej zewnętrznej instancji.

Proces trapper

Odbieranie żądań metryk wewnętrznych z zewnętrznej instancji Zabbix jest obsługiwane przez proces trapper, który weryfikuje żądanie, zbiera metryki, tworzy bufor danych JSON i odsyła przygotowany JSON z powrotem, na przykład z serwera:

```
{
  "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
      }
    }
  }
}
```

Pozycje wewnętrzne kolejki

Istnieją również dwie inne pozycje, które umożliwiają zdalne odpytywanie statystyk wewnętrznej kolejki w innej instancji Zabbix:

- `zabbix[stats,<ip>,<port>,queue,<from>,<to>]` pozycja wewnętrzna - do bezpośredniego odpytywania wewnętrznej

kolejki zdalnego serwera/proxy Zabbix
- `zabbix.stats[<ip>,<port>,queue,<from>,<to>]` pozycja agent - do odpytywania wewnętrznej kolejki zdalnego serwera/proxy Zabbix za pośrednictwem agenta

Zobacz także: Pozycje wewnętrzne, Pozycje Zabbix agent

Konfigurowanie Kerberosa z Zabbix

Przegląd

Uwierzytelnianie Kerberos może być używane w monitorowaniu WWW oraz w pozycjach HTTP w Zabbix.

Ta strona opisuje przykład konfiguracji Kerberos dla serwera Zabbix, aby wykonywać monitorowanie WWW `www.example.com` z użyciem principal Kerberos dla procesu Zabbix w Debian/Ubuntu.

Konfiguracja

1. Zainstaluj KDC i narzędzia klienta:

```
sudo apt update
sudo apt install krb5-kdc krb5-admin-server krb5-user
```

Podczas konfiguracji pakietu odpowiedz na pytania, na przykład:

```
Domyślna domena Kerberos w wersji 5: EXAMPLE.COM
Serwery Kerberos dla domeny: localhost (lub FQDN)
Serwer administracyjny dla domeny Kerberos: localhost (lub FQDN)
```

2. Zmapuj przyjazną nazwę hosta (opcjonalnie, na potrzeby testów lokalnych).

Edytuj plik /etc/hosts i dodaj wpis dotyczący kontrolera domeny oraz serwera WWW, jeśli nie masz DNS:

```
sudo vi /etc/hosts
```

Przykładowy wpis:

```
192.168.1.100  dc01.example.com dc01
```

3. Skonfiguruj klienta Kerberos i domenę KDC:

```
sudo vi /etc/krb5.conf
```

Przykładowe ustawienia:

```
[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
```

Jeśli planujesz używać `.localdomain` lub innych nazw niepublicznych, dodaj jawne mapowania domena→realm, aby mapowanie nazwa hosta→realm działało. Niezgodności w tym miejscu powodują błędy `Server not found in Kerberos database`.

4. Zainicjuj bazę danych Kerberos (jednorazowo, na hoście KDC). Po wyświetleniu monitu ustaw bezpieczne hasło główne:

```
sudo krb5_newrealm
```

5. Utwórz principal `HTTP/host.fqdn@REALM`, używając dokładnej nazwy hosta, której będą używać klienci; preferowane są małe litery (np. `HTTP/web.example.com@EXAMPLE.COM`). Niezgodność wielkości liter lub nazwy powoduje błąd `Server not found in Kerberos database`.

```
sudo kadmin.local
```

W `kadmin.local`:

```
addprinc kerb-admin@EXAMPLE.COM      # principal administracyjny
addprinc -randkey HTTP/dc01.example.com@EXAMPLE.COM
ktadd -k /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
quit
```

Przenieś keytab na hosta WWW (lub pozostaw go lokalnie, jeśli jest to ta sama maszyna) i ustaw uprawnienia umożliwiające korzystanie z niego przez Apache:

```
chown www-data:www-data /etc/apache2/http.keytab
chmod 600 /etc/apache2/http.keytab
#### weryfikacja
sudo -u www-data -k /etc/apache2/http.keytab
```

6. Zainstaluj i włącz moduł Apache GSSAPI:

```
sudo apt install libapache2-mod-auth-gssapi
sudo a2enmod auth_gssapi
sudo a2enmod headers
sudo systemctl restart apache2
```

> **Attention:**
> Nie wszystkie wersje mod_auth_gssapi obsługują każdą dyrektywę `Gssapi*`. Jeśli Apache zakończy działanie z błędem `Invalid command 'GssapiCredStore'`, usuń nieobsługiwaną dyrektywę lub zaktualizuj moduł.

7. Skonfiguruj VirtualHost (dostosuj `DocumentRoot` / ścieżkę do interfejsu Zabbix):

```
sudo vi /etc/apache2/sites-available/zabbix.conf
```

W pliku `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>
```

Uruchom ponownie Apache:

```
sudo systemctl restart apache2
```

8. Włącz/uruchom usługi KDC i sprawdź nasłuchiwanie na portach (host KDC):

```
sudo systemctl enable --now krb5-kdc krb5-admin-server
ss -tnlp | grep :80    # lub: sudo netstat -tnlp | grep :80
```

9. Uzyskaj TGT do testów (uruchom jako użytkownik, który będzie używać biletu).

Na liście biletów powinien być widoczny `krbtgt/EXAMPLE.COM@EXAMPLE.COM`. Uruchom `kinit` jako ten sam użytkownik systemu operacyjnego, który potrzebuje biletu (np. `zabbix` na potrzeby kontroli WWW lub `www-data`/Apache na potrzeby interaktywnych testów SSO w przeglądarce). Bilety wystawione dla innego użytkownika systemu operacyjnego nie będą widoczne, chyba

że zostaną dostosowane KRB5CCNAME i uprawnienia.

```
kinit kerb-admin@EXAMPLE.COM
klist
```

10. Przetestuj wymianę SPNEGO za pomocą curl (z klienta z prawidłowym TGT). Kod 200 OK (lub przekierowanie do aplikacji) oznacza pomyślne zakończenie SPNEGO:

```
curl -v --negotiate -u : http://dc01.example.com/
```

11. Opcjonalnie, jeśli interfejs Zabbix powinien akceptować logowania uwierzytelniane przez HTTP, włącz uwierzytelnianie HTTP w frontendzie Zabbix.

W pliku konfiguracyjnym frontendu (zabbix.conf.php) ustaw:

```
$ZBX_FEATURE_FLAGS['http_auth_enabled'] = true;
```

Następnie skonfiguruj uwierzytelnianie HTTP w interfejsie WWW.

12. Konfiguracja przeglądarki (jako przykład użyto Firefoksa): ustaw network.negotiate-auth.trusted-uris na hosta (hosty) wykonującego Negotiate (dc01.example.com), aby przeglądarka automatycznie wysyłała tokeny Kerberos.

W about:config:

```
network.negotiate-auth.trusted-uris = dc01.example.com
```

Teraz przejście do http://dc01.example.com powinno spowodować bezpośrednie zalogowanie do Zabbix, bez wyświetlania formularza.

13. Dbaj o aktualność kluczy i biletów. Domyślny czas życia biletu Kerberos wynosi około 10 godzin. Dodaj zadanie cron lub timer systemd, aby uniknąć wygaśnięcia:

```
#### dla usługi WWW
kinit -kt /etc/apache2/http.keytab HTTP/dc01.example.com@EXAMPLE.COM
#### dla użytkownika monitoringu
kinit -kt /var/lib/zabbix/kerb.keytab kerb-admin@EXAMPLE.COM
```

14. Kontrole porządkowe:

- klist -k /etc/apache2/http.keytab - sprawdź, czy principal usługi znajduje się w keytab.
- sudo tail -f /var/log/apache2/error.log - obserwuj błędy GSSAPI (gss_acquire_cred[_from]() failed to get server creds oznacza problem z keytabem/uprawnieniami lub brakujący principal).
- curl --negotiate zwracający 401/403 często oznacza nieprawidłowy principal, brak biletu, niezgodność nagłówka hosta lub problem z uprawnieniami systemu plików; sprawdź logi oraz mapowania domen w /etc/krb5.conf.

Uwagi dotyczące bezpieczeństwa i uprawnień do plików

Pliki keytab muszą być dostępne do odczytu wyłącznie dla konta, które ich potrzebuje. Przykładowe uprawnienia: 0400 z właścicielem zabbix:zabbix w przypadku keytab użytkownika zabbix albo 0440 z właścicielem root:www-data w przypadku keytab Apache.

Unikaj przechowywania na hoście haseł w postaci jawnego tekstu, które zachowują ważność przez długi czas. W miarę możliwości używaj keytabów lub principalów maszyn dołączonych do domeny.

Podczas uruchamiania testów lub skryptów, które ustawiają KRB5CCNAME albo kopiują keytaby, po zakończeniu operacji ponownie sprawdź właściciela i uprawnienia — odrzucanie poświadczeń przez serwer WWW jest często spowodowane problemem z uprawnieniami do pliku.

Parametry modbus.get

Przegląd

Poniższa tabela przedstawia szczegóły parametrów pozycji modbus.get.

Parametry

Parameter	Opis	Defaults	Example
endpoint	Protokół i adres endpointu, zdefiniowane jako `protocol://connection_string` Możliwe wartości protokołu: *rtu*, *ascii* (tylko Agent 2), *tcp* Format connection string: dla *tcp* - `address:port` dla linii szeregowej: *rtu*, *ascii* - `port_name:speed:params` gdzie 'speed' - 1200, 9600 itd. 'params' - bity danych (5,6,7 lub 8), parzystość (n,e lub o dla brak/parzysta/nieparzysta), bity stopu (1 lub 2)	protocol: brak protokół *rtu/ascii*: port_name: brak speed: 115200 params: 8n1 protokół *tcp*: address: brak port: 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	Adres Modbus urządzenia, do którego jest przeznaczony (1 do 247), zobacz MODBUS Messaging Implementation Guide (strona 23) urządzenie tcp (nie GW) zignoruje to pole	serial: 1 tcp: 255 (0xFF)	2
function	Puste lub wartość obsługiwanej funkcji: 1 - Odczyt cewki, 2 - Odczyt wejścia dyskretnego, 3 - Odczyt rejestrów przechowujących, 4 - Odczyt rejestrów wejściowych	puste	3
address	Adres pierwszego rejestru, cewki lub wejścia. Jeśli 'function' jest puste, wtedy 'address' powinien należeć do zakresu dla: Cewka - 00001 - 09999 Wejście dyskretne - 10001 - 19999 Rejestr wejściowy - 30001 - 39999 Rejestr przechowujący - 40001 - 49999 Jeśli 'function' nie jest puste, pole 'address' będzie miało zakres od 0 do 65535 i zostanie użyte bez modyfikacji (PDU)	pusta function: 00001 niepusta function: 0	9999
count	Liczba sekwencyjnych wartości 'type', które zostaną odczytane z urządzenia, gdzie: dla cewki lub wejścia dyskretnego 'type' = 1 bit dla pozostałych przypadków: (count*sizeof(type))/2 = rzeczywista liczba rejestrów do odczytu Jeśli 'offset' nie jest równe 0, wartość ta zostanie dodana do 'rzeczywistej liczby' Dopuszczalny zakres dla 'rzeczywistej liczby' to 1:65535	1	2

Parameter	Opis	Defaults	Example
type	Typ danych: dla Read Coil i Read Discrete Input - *bit* dla Read Holding Registers i Read Input Registers: *int8* - 8bit *uint8* - 8bit (bez znaku) *int16* - 16bit *uint16* - 16bit (bez znaku) *int32* - 32bit *uint32* - 32bit (bez znaku) *float* - 32bit *uint64* - 64bit (bez znaku) *double* - 64bit	bit uint16	uint64
endianness	Typ kolejności bajtów: *be* - Big Endian *le* - Little Endian *mbe* - Mid-Big Endian *mle* - Mid-Little Endian Ograniczenia: dla 1 bitu - be dla 8 bitów - be,le dla 16 bitów - be,le	be	le
offset	Liczba rejestrów, począwszy od 'address', których wynik zostanie odrzucony. Rozmiar każdego rejestru to 16bit (potrzebne do obsługi urządzeń, które nie obsługują losowego dostępu do odczytu).	0	4

Tworzenie niestandardowych nazw liczników wydajności dla VMware

Przegląd

Ścieżka licznika wydajności VMware ma format `group/counter[rollup]`, gdzie:

- `group` - grupa licznika wydajności, na przykład *cpu*
- `counter` - nazwa licznika wydajności, na przykład *usagemhz*
- `rollup` - typ agregacji licznika wydajności, na przykład *average*

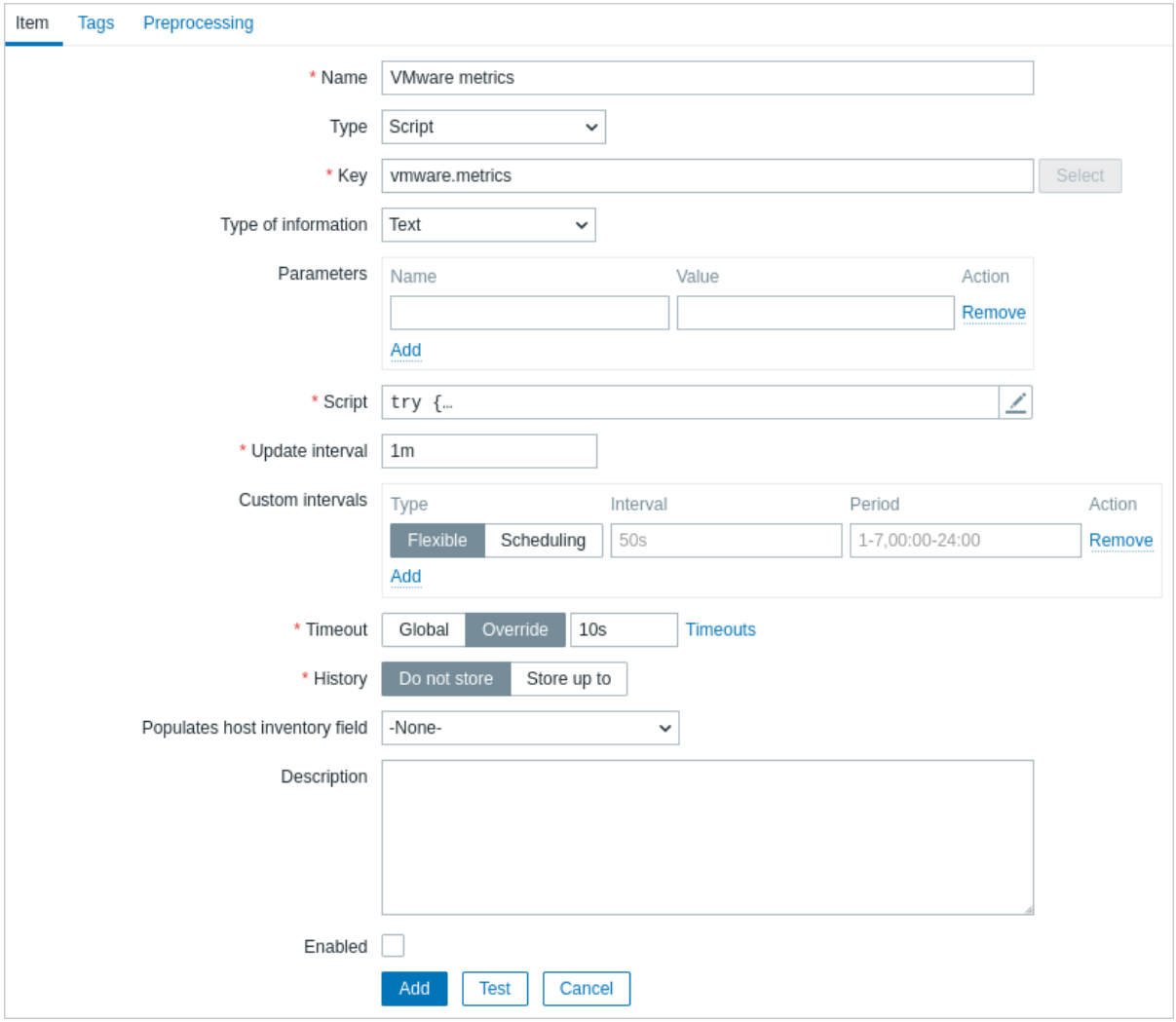
Zatem powyższy przykład da następującą ścieżkę licznika: `cpu/usagemhz[average]`

Opisy grup liczników wydajności, nazwy liczników oraz typy agregacji można znaleźć w dokumentacji VMware.

Możliwe jest uzyskanie nazw wewnętrznych i tworzenie niestandardowych nazw liczników wydajności za pomocą elementu typu script w Zabbix.

Konfiguracja

1. Utwórz wyłączoną pozycję Script na głównym hoście VMware (na którym znajduje się pozycja **eventlog[]**) z następującymi parametrami:



- *Name*: VMware metrics
- *Type*: Script
- *Key*: vmware.metrics
- *Type of information*: Text
- *Script*: skopiuj i wklej poniższy skrypt
- *Timeout*: 10
- *History*: Do not store
- *Enabled*: unmarked

Skrypt

```
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;
```

Po skonfigurowaniu pozycji kliknij przycisk *Test*, a następnie *Get value*.



Skopiuj otrzymany XML do dowolnego formatera XML i znajdź żądaną metrykę.

Przykład XML dla jednej metryki:

```
<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>
```

Użyj XPath, aby wyodrębnić ścieżkę licznika z otrzymanego XML. W powyższym przykładzie XPath będzie następujący:

field	xPath	value
group	//groupInfo[../key=6]/key	cpu
counter	//nameInfo[../key=6]/key	usagemhz
rollup	//rollupType[../key=6]	average

W tym przypadku wynikowa ścieżka licznika wydajności to: `cpu/usagemhz[average]`

Wartości zwracane dla system.sw.packages.get

Przegląd

Ta sekcja zawiera szczegółowe informacje o wartościach zwracanych dla pozycji agenta Zabbix `system.sw.packages.get`.

Szczegóły

Dane wyjściowe tej pozycji to tablica obiektów, z których każdy zawiera następujące właściwości:

- **name** - nazwa pakietu
- **manager** - menedżer pakietów, który zgłosił te dane (`rpm`, `dpkg`, `pacman`, `pkgtool` lub `portage`)
- **version** - wersja pakietu
- **size** - rozmiar nieskompresowanego pakietu w bajtach (jeśli niedostępny, ustaw na 0)
- **arch** - architektura pakietu
- **buildtime** - obiekt z 2 wpisami:
  - **timestamp** - znacznik czasu UNIX określający, kiedy pakiet został zbudowany (jeśli niedostępny, ustaw na 0)
  - **value** - czytelna dla człowieka data i godzina zbudowania pakietu (jeśli niedostępna, ustaw na pusty ciąg znaków)
- **installtime** - obiekt z 2 wpisami:
  - **timestamp** - znacznik czasu UNIX określający, kiedy pakiet został zainstalowany (jeśli niedostępny, ustaw na 0)
  - **value** - czytelna dla człowieka data i godzina zainstalowania pakietu (jeśli niedostępna, ustaw na pusty ciąg znaków)

Przykład:

```
[
    {
        "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": ""
        }
    }
]
```

Zwracane wartości dla net.dns.get

Przegląd

Ta sekcja zawiera szczegółowe informacje o wartościach zwracanych dla pozycji Zabbix agent 2 `net.dns.get`.

Szczegóły

Wyjście tej pozycji jest obiektem zawierającym informacje o rekordach DNS na podstawie parametrów podanych w kluczu pozycji.

Na przykład pozycja `net.dns.get[,example.com]` może zwrócić następujący JSON dla odrzuconego zapytania:

```
{
    "flags": [
        "RA"
    ],
    "query_time": "0.019030",
    "question_section": [
        {
            "qclass": "IN",
            "qname": "example.com.",
            "qtype": "SOA"
        }
    ],
    "response_code": "REFUSED",
    "zbx_error_code": 0
}
```

Po podaniu adresu IP serwera DNS pozycja `net.dns.get[192.0.2.1,example.com]` może zwrócić następujący 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
}
```


Jeśli wystąpi problem z połączeniem, pozycja `net.dns.get[192.0.2.1,example.com]` może zwrócić błąd:


```
{
    "zbx_error_code": -1,
    "zbx_error_msg": "Communication error: read udp 192.0.2.1:12345->192.0.2.1:53: i/o timeout"
}
```


Możliwe są następujące typy kodów błędów:

Scenario	"zbx_error_code"	"zbx_error_msg"
Brak błędów, a odpowiedź DNS została odebrana i przeanalizowana.	0	
DNS jest niedostępny.	-1	"Communication error"
Błąd występuje podczas parsowania JSON	-2	"Received unexpected response"

Przy dodatkowych parametrach pozycja `net.dns.get[192.0.2.1,example.com,ANY,5,5,tcp,"cdflag,rdflag,dnssec,nsid,edn` może zwrócić następujący 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
}
```


Zobacz także

Aby uzyskać więcej informacji o rekordach DNS, zobacz:

- Domain Names - Implementation and Specification
- Domain Name System (DNS) Parameters

Zwracane wartości dla net.if.get

Przegląd

Ta sekcja zawiera szczegóły dotyczące wartości zwracanej dla pozycji `net.if.get`. Ta pozycja jest obsługiwana w systemie Linux przez agent Zabbix oraz agent Zabbix 2.

Szczegóły

Ta pozycja zwraca szczegółowe informacje o interfejsach sieciowych, których nazwy pasują do podanego wyrażenia regularnego. Dane wyjściowe mają strukturę obiektu JSON zawierającego tablice `config` i `values`.

Dane wyjściowe zawierają następujące właściwości:

- **name** - nazwa interfejsu, odczytywana z `/proc/net/dev`;
- **ifalias** - alias interfejsu, odczytywany z `/sys/class/net/<name>/ifalias`;
- **mac** - adres MAC interfejsu, odczytywany z `/sys/class/net/<name>/address`;

- **type** - typ interfejsu (`physical`, `virtual`, `loopback`);
- **speed** - prędkość łącza w Mbit/s, specyficzna dla Ethernet, odczytywana z `/sys/class/net/<name>/speed`;
- **duplex** - ostatnia lub bieżąca wartość dupleksu interfejsu (`half`, `full`), specyficzna dla Ethernet, odczytywana z `/sys/class/net/<name>/duplex`;
- **administrative_state** - skonfigurowany stan (`down`, `up`), odczytywany z `/sys/class/net/<name>/flags`;
- **operational_state** - stan operacyjny interfejsu RFC2863 (`unknown`, `notpresent`, `down`, `lowerlayerdown`, `testing`, `dormant`, `up`), odczytywany z `sys/class/net/<name>/operstate`;
- **carrier** - obecność fizycznego łącza (0 - fizyczne łącze jest wyłączone, 1 - fizyczne łącze jest włączone), odczytywana z `/sys/class/net/<name>/carrier`;
- **carrier_changes** - całkowita liczba zmian stanu łącza z up na down i odwrotnie, odczytywana z `/sys/class/net/<name>/carrier_c`
- **carrier_up_count** - liczba przypadków, gdy łącze przeszło do stanu up, odczytywana z `/sys/class/net/<name>/carrier_up_coun`
- **carrier_down_count** - liczba przypadków, gdy łącze przeszło do stanu down, odczytywana z `/sys/class/net/<name>/carrier_do`
- **in** - ruch przychodzący, odczytywany z `/proc/net/dev`. Zawiera liczniki interfejsu:
  - **bytes** - liczba poprawnych bajtów;
  - **packets** - liczba poprawnych pakietów;
  - **errors** - całkowita liczba błędnych pakietów;
  - **dropped** - liczba odebranych, ale nieprzetworzonych pakietów + liczba pakietów pominiętych przez host;
  - **overruns** - liczba zdarzeń przepełnienia;
  - **frame** - liczba pakietów z błędami ramek;
  - **compressed** - liczba poprawnie odebranych skompresowanych pakietów;
  - **multicast** - odebrane pakiety multicast.
- **out** - ruch wychodzący, odczytywany z `/proc/net/dev`. Zawiera liczniki interfejsu:
  - **bytes** - liczba poprawnych bajtów;
  - **packets** - liczba poprawnych pakietów;
  - **errors** - całkowita liczba błędnych pakietów;
  - **dropped** - liczba odebranych, ale nieprzetworzonych pakietów + liczba pakietów pominiętych przez host;
  - **overruns** - liczba zdarzeń przepełnienia;
  - **collisions** - liczba kolizji podczas transmisji pakietów;
  - **carrier** - liczba błędów transmisji ramek spowodowanych utratą nośnej podczas transmisji;
  - **compressed** - liczba wysłanych skompresowanych pakietów.

Niedostępne metryki są wykluczane z danych wyjściowych JSON. W poniższym przykładzie dla interfejsu „lo” nie skonfigurowano aliasu, dlatego właściwość `ifalias` jest wykluczona.

Przykład:


```
{
    "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
            }
        }
    ]
}
```


Zobacz także

Aby uzyskać więcej informacji o ABI sysfs dla urządzeń sieciowych, zobacz:

- Dokumentację jądra Linux

Wartości zwracane dla vfs.dev.get

Przegląd

Ta sekcja zawiera szczegóły wartości zwracanych dla pozycji `vfs.dev.get`. Pozycja jest obsługiwana w systemie Linux przez agent i Zabbix agent 2, a w systemie AIX przez agent.

Szczegóły

Ta pozycja zwraca szczegółowe informacje o urządzeniach blokowych, których nazwy pasują do podanego wyrażenia regularnego.

Dane wyjściowe są strukturyzowane jako obiekt JSON zawierający tablice `config` oraz, w zależności od parametru `<mode>`, `values`.

Tablica `config` zawiera właściwości dostępne dla wybranego trybu. Tablica `values` jest zwracana tylko dla trybów, które udostępniają statystyki urządzeń blokowych.

Niedostępne metryki są wykluczane z danych wyjściowych JSON.

W poniższych sekcjach opisano właściwości zwracane dla każdego trybu.

tryb dysków

`vfs.dev.get[disks,.*]` zwraca dane konfiguracyjne urządzeń blokowych.

Właściwości:

- **name** - nazwa urządzenia blokowego;
- **devid** - identyfikator urządzenia blokowego, na przykład identyfikator urządzenia oparty na WWN lub wygenerowany przez jądro;
- **type** - typ urządzenia blokowego (`disk`, `partition`, `rom`);
- **path** - ścieżka do węzła urządzenia blokowego;
- **model** - model urządzenia;

- **serial** - numer seryjny urządzenia;
- **wwn** - World Wide Name (WWN) urządzenia;
- **size_bytes** - rozmiar urządzenia w bajtach;
- **logical_block_size** - logiczny rozmiar bloku w bajtach;
- **physical_block_size** - fizyczny rozmiar bloku w bajtach.

Przykład dla 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
    }
  ]
}
```


Przykład dla AIX:


```
{
  "config": [
    {
      "name": "cd0",
      "type": "disk",
      "path": "/dev/cd0",
      "size_bytes": 0,
      "logical_block_size": 512
    },
    {
      "name": "hdisk1",
      "type": "disk",
      "devid": "48333321360050768 0C818022F000000000015EA404214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk1",
      "size_bytes": 51136000000,
      "logical_block_size": 512
    },
```

```
    {
      "name": "hdisk0",
      "type": "disk",
      "devid": "48333213600507680C818022F000000000015EA304214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk0",
      "size_bytes": 0,
      "logical_block_size": 512
    }
  ]
}
```

tryb urządzeń

`vfs.dev.get[devices,.*]` zwraca dane konfiguracyjne urządzeń blokowych, w tym partycje.

Właściwości:

- **name** - nazwa urządzenia blokowego.
- **devid** - identyfikator urządzenia blokowego.
- **type** - typ urządzenia blokowego (`disk`, `partition`, `rom`).
- **partitions** - partycje na urządzeniu. Zawiera nazwy partycji oraz ich rozmiary w bajtach.

Przykład dla 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
      }
    }
  ]
}
```

tryb disk_stats

`vfs.dev.get[disk_stats,.*]` zwraca dane konfiguracyjne urządzeń blokowych oraz statystyki dysków.

Właściwości:

- **name** - nazwa urządzenia blokowego.
- **devid** - identyfikator urządzenia blokowego.

- **type** - typ urządzenia blokowego (`disk`, `partition`, `rom`).
- **size_bytes** - rozmiar urządzenia w bajtach.
- **logical_block_size** - logiczny rozmiar bloku w bajtach.
- **stats** - statystyki I/O urządzenia blokowego. Zawiera:
  - **reads_completed** - liczba zakończonych operacji odczytu;
  - **writes_completed** - liczba zakończonych operacji zapisu;
  - **bytes_read** - liczba odczytanych bajtów;
  - **bytes_written** - liczba zapisanych bajtów;
  - **io_time_ms** - łączny czas spędzony na operacjach I/O, w milisekundach.

Przykład dla 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
      }
    }
```

```
  ]
}
```

Przykład dla AIX:

```
{
  "config": [
    {
      "name": "cd0",
      "type": "disk",
      "path": "/dev/cd0",
      "size_bytes": 0,
      "logical_block_size": 2048
    },
    {
      "name": "hdisk1",
      "type": "disk",
      "devid": "48333321360050768 0C818022F000000000015EA404214503IBMfcp05VDASD03AIXvscsi",
      "path": "/dev/hdisk1",
      "size_bytes": 51136000000,
      "logical_block_size": 512
    },
    {
      "name": "hdisk0",
      "type": "disk",
      "devid": "48333321360050768 0C818022F000000000015EA304214503IBMfcp05VDASD03AIXvscsi",
      "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
      }
    }
  ]
```

```
}
```

tryb device_stats

`vfs.dev.get[device_stats,.*]` zwraca dane konfiguracji urządzeń blokowych oraz statystyki urządzeń.

Właściwości:

- **name** - nazwa urządzenia blokowego.
- **devid** - identyfikator urządzenia blokowego.
- **type** - typ urządzenia blokowego (`disk`, `partition`, `rom`).
- **size_bytes** - rozmiar urządzenia w bajtach.
- **stats** - statystyki I/O urządzenia blokowego. Zawiera:
  - **reads_completed** - liczba zakończonych operacji odczytu;
  - **writes_completed** - liczba zakończonych operacji zapisu;
  - **bytes_read** - liczba odczytanych bajtów;
  - **bytes_written** - liczba zapisanych bajtów;
  - **io_time_ms** - łączny czas spędzony na operacjach I/O, w milisekundach.

Przykład dla 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
      }
    }
  ]
}
```

Uwagi dotyczące pozycji system.cpu.util w systemie Windows

system.cpu.util pozycja dostarcza procent wykorzystania CPU.

Gdy w Zabbix agent dla Windows uruchamiany jest proces collectora, inicjalizowany jest bufor dla N pozycji dla N procesorów logicznych (wątków). Wartości są okresowo aktualizowane w buforze przy użyciu liczników wydajności Windows. Wartości te są zwracane, gdy żądane jest `system.cpu.util[n]`, gdzie 0 <= n < N.

Jeszcze jedna pozycja w buforze jest używana dla `system.cpu.util[total]`.

Zabbix automatycznie wybiera różne liczniki wydajności, aby obsługiwać zarówno systemy NUMA, jak i starsze wersje Windows bez obsługi grup procesorów.

Jeśli liczba procesorów logicznych (wątków) jest mniejsza lub równa 64, a liczba grup procesorów jest równa 1, wtedy:

```
N to liczba procesorów logicznych (wątków).

\Processor(_Total)\% Processor Time
\Processor(0)\% Processor Time
\Processor(1)\% Processor Time
\Processor(2)\% Processor Time
...
\Processor(N-1)\% Processor Time
```

W przeciwnym razie:

```
G to liczba grup procesorów, a N to liczba procesorów logicznych (wątków) w grupie.

\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
```

Obsługa dużych wartości JSON

Aby zapewnić niezawodne przetwarzanie dużych wartości z typem danych JSON data type (1 MiB lub więcej), przejrzyj i zastosuj poniższe zmiany konfiguracji w razie potrzeby:

1. Jeśli używasz MySQL lub MariaDB, zwiększ maksymalny dozwolony rozmiar pakietu w pliku `/etc/mysql/my.cnf` (na niektórych systemach może domyślnie wynosić 1 MB):

```
[mysqld]
max_allowed_packet=128M

[mysqldump]
max_allowed_packet=1G
```

2. W razie potrzeby dostosuj dodatkowe server system variables MySQL/MariaDB oraz InnoDB system variables. Na przykład:

```
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. Zwiększ wartość `HistoryCacheSize` do 2G w pliku konfiguracyjnym serwer Zabbix.

4. Jeśli używasz Zabbix proxy, zwiększ wartość `HistoryCacheSize` do 2G oraz `ProxyMemoryBufferSize` do 2G w pliku konfiguracyjnym proxy.

5. Jeśli używasz Zabbix agent, zwiększ limity czasu komunikacji między serwerem/proxy Zabbix a agentem (na przykład item timeouts lub wartość parametru `Timeout` w plikach konfiguracyjnych Zabbix server/proxy i agent). W przeciwnym razie wiadomości między nimi mogą nie zostać w pełni przesłane i mogą pojawić się błędy takie jak *"message length does not match expected length"*. W razie potrzeby przejrzyj także inne limity czasu, takie jak limit czasu script timeout lub limit czasu item test.

Obsługiwane funkcje

Zobacz Wyrażenie wyzwalacza i funkcje.

Makra

Możliwe jest użycie gotowych obsługiwanych makr oraz makr użytkownika obsługiwanych według lokalizacji.

Makra obsługiwane przez lokalizację

Przegląd

Tabela makr obsługiwanych według lokalizacji zawiera zestawienie wszystkich makr (wbudowanych, wyrażeń i definiowanych przez użytkownika), które są obsługiwane w określonej lokalizacji.

Jeśli wszystkie makra z powiązanej grupy są obsługiwane, podawana jest tylko nazwa grupy. Kliknij nazwę każdego makra lub nazwę grupy, aby zobaczyć pełne szczegóły dotyczące makr.

Lista szczegółów makr zawiera szczegółowe zestawienie wszystkich makr **wbudowanych**, pogrupowanych według obszaru zastosowania. Zwróć uwagę, że aby dostosować wartości makr (na przykład skrócić je lub wyodrębnić określone podciągi), możesz użyć funkcji makr.

Makra obsługiwane przez lokalizację

Lokalizacja	Obsługiwane makra	Komentarze
Action		
Warunek okresu czasu, Domyślna długość kroku operacji, Długość kroku	Definiowane przez użytkownika makra (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane
Powiadomienia i polecenia		
Oparte na wyzwalaczu	Makra Action Makra Data/godzina {ESC.HISTORY} Makra Zdarzenie, makra Zdarzenie przyczyna/objaw Makra aktualizacji zdarzenia: {EVENT.UPDATE.HISTORY}, {EVENT.UPDATE.STATUS} Makro wyrażenia: {?EXPRESSION} {FUNCTION.VALUE} Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra host docelowy (tylko polecenia) Makra inwentarza hosta Makra pozycja (z wyjątkiem {ITEM.STATE}, {ITEM.STATE.ERROR}) Makra proxy Makra wyzwalacza: {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} Definiowane przez użytkownika makra	Zobacz też: Używanie makr w wiadomościach

Lokalizacja	Obsługiwane makra	Komentarze
Aktualizacja problemu	Makra Action Makra Data/godzina {ESC.HISTORY} Makra Zdarzenie, makra Zdarzenie przyczyna/objaw, makra Zdarzenie odzyskania Makra aktualizacji zdarzenia: wszystkie (z wyjątkiem {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}) Makro wyrażenia: {?EXPRESSION} Makra funkcji Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra host docelowy (tylko polecenia) Makra inwentarza hosta Makra pozycja (z wyjątkiem {ITEM.STATE}, {ITEM.STATE.ERROR}) Makra proxy Makra wyzwalacza: {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} Definiowane przez użytkownika makra	
Odzyskanie problemu	Makra Zdarzenie odzyskania Makra Zdarzenie {FUNCTION.RECOVERY.VALUE} {TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}	
Odkrywanie	Makra Action Makra Data/godzina Makra Odkrywanie Makra zdarzenia: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP} Makra host docelowy (tylko polecenia) Makra proxy	
Autorejestracja	Makra Action Makra Data/godzina Makra zdarzenia: {EVENT.AGE}, {EVENT.DATE}, {EVENT.ID}, {EVENT.OBJECT}, {EVENT.SOURCE}, {EVENT.TIME}, {EVENT.TIMESTAMP} Makra hosta: {HOST.HOST}, {HOST.IP}, {HOST.PORT}, {HOST.METADATA} Makra host docelowy (tylko polecenia) Makra proxy	
Oparte na usługach	Makra Action Makra Data/godzina {ESC.HISTORY} Makra zdarzenia: wszystkie z wyjątkiem {EVENT.ACK.STATUS}, {EVENT.OPDATA} Makra usługi Definiowane przez użytkownika makra	
Odzyskanie usługi	Makra Zdarzenie (z wyjątkiem {EVENT.ACK.STATUS}, {EVENT.OPDATA}), makra Zdarzenie odzyskania	

Lokalizacja	Obsługiwane makra	Komentarze
Aktualizacja usługi	Makra Action Makra Data/godzina {ESC.HISTORY} Makra Zdarzenie (z wyjątkiem {EVENT.ACK.STATUS}, {EVENT.OPDATA}) Makra aktualizacji zdarzenia: {EVENT.UPDATE.DATE}, {EVENT.UPDATE.NSEVERITY}, {EVENT.UPDATE.SEVERITY}, {EVENT.UPDATE.STATUS}, {EVENT.UPDATE.TIME}, {EVENT.UPDATE.TIMESTAMP} Makra usługi Definiowane przez użytkownika makra	
Powiadomienia wewnętrzne	Action macros Date/time macros {ESC.HISTORY} Event macros (except {EVENT.NSEVERITY}, {EVENT.SEVERITY}), Recovery event macros Host macros: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Host inventory macros Item macros: {ITEM.DESCRIPTION}, {ITEM.DESCRIPTION.ORIG}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, {ITEM.NAME}, {ITEM.NAME.ORIG}, {ITEM.VALUETYPE} Proxy macros	
Oparte na pozycji	Makra pozycji: {ITEM.STATE}, {ITEM.STATE.ERROR}	
Oparte na regule LLD	Makra powiadomienia o wykrywaniu niskiego poziomu	
Oparte na wyzwalaczu	Makra wyzwalacza: {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} Definiowane przez użytkownika makra	
Parametry skryptu alarmu	Makra Alert Definiowane przez użytkownika makra (tylko globalne)	
Łącznik		
Pole URL, Nazwa użytkownika, Hasło, Token Bearer, HTTP proxy, certyfikat SSL, plik klucza SSL, hasło klucza SSL	Definiowane przez użytkownika makra (tylko globalne)	
Pole Timeout	Definiowane przez użytkownika makra (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.

Lokalizacja	Obsługiwane makra	Komentarze
Nazwa wykresu	Makro wyrażenia: {?EXPRESSION}	W tym makrze w tej lokalizacji obsługiwana jest tylko pojedyncza funkcja **avg**, **last**, **max** lub **min** z parametrem w sekundach (można użyć sufiksów czasu). Makro {HOST.HOST<1-9>} może być użyte jako host wewnątrz makra (zobacz też makra indeksowane). Przykład użycia: {?avg(/{HOST.HOST}/item.key,1h)} Nieobsługiwane użycie: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)}
Host, prototyp hosta		
Adres IP/DNS interfejsu	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME} Definiowane przez użytkownika makra [1]	[1] Dla adresu IP pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane
Port interfejsu	Definiowane przez użytkownika makra	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Społeczność SNMP *Nazwa kontekstu SNMPv3, nazwa bezpieczeństwa, hasło uwierzytelniania, hasło prywatności* *Nazwa użytkownika* *IPMI, hasło*	Definiowane przez użytkownika makra	
Pozycja, prototyp pozycji, reguła LLD		
Nazwa	Definiowane przez użytkownika makra	Nieobsługiwane w nazwach reguł LLD.
Opis	Definiowane przez użytkownika makra	

Lokalizacja	Obsługiwane makra	Komentarze
Parametry klucza	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Definiowane przez użytkownika makra	Obsługiwane makra {HOST.*} zostaną rozwiązane do interfejsu wybranego dla pozycji. Gdy są używane w pozycjach bez interfejsów, zostaną rozwiązane do interfejsu agenta Zabbix, SNMP, JMX lub IPMI hosta, w tej kolejności priorytetu, albo do wartości 'UNKNOWN', jeśli host nie ma żadnego interfejsu. Makra nie są obsługiwane w kluczu pozycji typu Zabbix trapper.
Interwał aktualizacji, *Interwały niestandardowe*, *Timeout (dostępne dla obsługiwanych typów pozycji)*, *Przechowuj do (dla historii/trendów)* pola	Definiowane przez użytkownika makra	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Usuń utracone zasoby, *Wyłącz utracone zasoby* pola	Definiowane przez użytkownika makra	Te pola są obsługiwane tylko dla reguł LLD. Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Parametry kroku preprocesowania (w tym skrypty niestandardowe), *niestandardowe parametry obsługi błędów* (pola *Ustaw wartość na* i *Ustaw błąd na*)	Definiowane przez użytkownika makra	
Wyrażenia regularne filtra	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Definiowane przez użytkownika makra	Te pola są obsługiwane tylko dla reguł LLD.
Wyrażenia regularne filtra zastąpienia	Definiowane przez użytkownika makra	Te pola są obsługiwane tylko dla reguł LLD.
Operacje zastąpienia: *Interwał aktualizacji*, *Okres przechowywania historii*, *Okres przechowywania trendów* (dla prototypów pozycji)	Definiowane przez użytkownika makra	Te pola są obsługiwane tylko dla reguł LLD. Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Typy pozycji		

Lokalizacja	Obsługiwane makra	Komentarze
Browser pozycja *nazwy parametrów i wartości*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} Definiowane przez użytkownika makra	
Script pozycja *nazwy parametrów i wartości*		
Obliczana/agregowana formuła pozycji (*stałe wyrażenia*, *parametry funkcji*, *parametry klucza pozycji*, (tylko pozycja agregowana) *warunki filtra* (nazwa grupy hostów i nazwa tagu))	Definiowane przez użytkownika makra	
Database monitor pozycja *pola Nazwa użytkownika, Hasło, zapytanie SQL*	Definiowane przez użytkownika makra	
HTTP agent pozycja *pola URL, pola zapytania, treść żądania, nagłówki (nazwy i wartości), plik certyfikatu SSL, plik klucza SSL*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} Definiowane przez użytkownika makra	Adresy URL zawierające tajne makro nie będą działać, ponieważ makro w nich zostanie rozwiązane jako ”******”
HTTP agent pozycja *pole HTTP Proxy*	Makra pozycji: {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} Definiowane przez użytkownika makra	
HTTP agent pozycja *pola Wymagane kody statusu, Nazwa użytkownika/hasło uwierzytelniania HTTP, hasło klucza SSL*	Definiowane przez użytkownika makra	
HTTP agent pozycja *pole Dozwolone hosty*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME} Definiowane przez użytkownika makra	
JMX agent pozycja *pole Endpoint*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.PORT} Definiowane przez użytkownika makra	
Trapper pozycja *pole Dozwolone hosty*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP} Definiowane przez użytkownika makra	
SNMP agent pozycja *pole SNMP OID*	Definiowane przez użytkownika makra	
SSH and Telnet pozycja *skrypt*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Definiowane przez użytkownika makra	
SSH agent pozycja *pola Nazwa użytkownika, Plik klucza publicznego, Plik klucza prywatnego, Hasło*	Definiowane przez użytkownika makra	
Telnet agent pozycja *pola Nazwa użytkownika, Hasło*	Definiowane przez użytkownika makra	
Mapa		

Lokalizacja	Obsługiwane makra	Komentarze
URL, nazwa URL	Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} {HOSTGROUP.ID} Makra inwentarza hosta Makra mapy {TRIGGER.ID}	
Etykieta elementu	Makro wyrażenia: {?EXPRESSION} [1] Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} {HOSTGROUP.ID} Makra inwentarza hosta Makra mapy Makra wyzwalacza: {TRIGGER.EVENTS.*}, {TRIGGER.ID}, {TRIGGER.PROBLEM.EVENTS.*}, {TRIGGERS.ACK}, {TRIGGERS.PROBLEM.ACK}, {TRIGGERS.PROBLEM.UNACK}, {TRIGGERS.UNACK}	[1] W tej lokalizacji w tym makrze obsługiwana jest tylko pojedyncza funkcja **avg**, **last**, **max** lub **min** z parametrem w sekundach (można użyć sufiksów czasu). Makro {HOST.HOST<1-9>} może być użyte jako host wewnątrz makra. Przykład użycia: {?avg(/{HOST.HOST}/item.key,1h)} Nieobsługiwane użycie: {?last(/host/item1)/last(/host/item2)} {?last(/host/item1)*10} {?count(/host/item1,5m)}
Etykieta łącza	Makro wyrażenia: {?EXPRESSION} [1]	
Etykieta kształtu *Pole tekstowe w kształtach mapy*	Makro wyrażenia: {?EXPRESSION} [1] {MAP.NAME}	
Wykrywanie sieci		
Pole Interwał aktualizacji	Definiowane przez użytkownika makra (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Pola Społeczność SNMP, SNMP OID *Pola Nazwa kontekstu SNMPv3, nazwa bezpieczeństwa, hasło uwierzytelniania, hasło prywatności*	Definiowane przez użytkownika makra (tylko globalne)	
Proxy		
Adres dla aktywnych agentów > *Port* (jeśli proxy należy do grupy) W trybie pasywnego proxy: pola interfejsu *Adres* i *Port* Pola *Timeouty dla typów pozycji* w trybie zastąpienia	Definiowane przez użytkownika makra (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Grupa proxy		
Okres awaryjnego przełączenia, *Minimalna liczba proxy* pola	Definiowane przez użytkownika makra (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.

Lokalizacja	Obsługiwane makra	Komentarze
Skrypty		
Polecenia, tekst potwierdzenia (Typ: skrypt, ręczna akcja **hosta**)	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID} [1], {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta [2] {MANUALINPUT} Makra Username Definiowane przez użytkownika makra	[1] Tylko tekst potwierdzenia [2] Obsługiwane dla serwera Zabbix i proxy Zabbix
Polecenia (Typ: skrypt, ręczna akcja **zdarzenia**)	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycja (z wyjątkiem {ITEM.STATE}, {ITEM.STATE.ERROR}) {MANUALINPUT} Makra proxy Makra wyzwalacza: {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} Makra Username Definiowane przez użytkownika makra	
Tekst potwierdzenia (Typ: skrypt, ręczna akcja **zdarzenia**)	Makra zdarzenia: {EVENT.ID}, {EVENT.NAME}, {EVENT.NSEVERITY}, {EVENT.SEVERITY}, {EVENT.STATUS}, {EVENT.VALUE} Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta {MANUALINPUT} Makra Username Definiowane przez użytkownika makra	
URL, tekst potwierdzenia (Typ: URL, ręczna akcja **hosta/zdarzenia**)		
Nazwa i wartość tagu	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra wyzwalacza: {TRIGGER.ID} (tylko wartość tagu wyzwalacza) Definiowane przez użytkownika makra	Makra nazwy i wartości tagu są rozwijane tylko podczas procesu generowania zdarzenia.
Trigger		
Nazwa	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra/referencje pozycyjne: $1 . . . $9 Definiowane przez użytkownika makra	

Lokalizacja	Obsługiwane makra	Komentarze
Nazwa zdarzenia	Makra Data/godzina: {TIME}, {TIMESTAMP} Makro wyrażenia: {?EXPRESSION} Makra funkcji Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Makra wyzwalacza: {TRIGGER.EXPRESSION.EXPLAIN}	
Dane operacyjne	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Definiowane przez użytkownika makra	
Wyrażenie	{TRIGGER.VALUE} Definiowane przez użytkownika makra [1]	[1] Tylko w stałych i parametrach funkcji; tajne makra nie są obsługiwane
Pola Tag do dopasowania, Nazwa wpisu menu, URL wpisu menu	Definiowane przez użytkownika makra	Adresy URL zawierające tajne makro nie będą działać, ponieważ makro w nich zostanie rozwiązane jako "******".
Opis	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} Definiowane przez użytkownika makra	
URL	{EVENT.ID} Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra pozycji: {ITEM.LASTVALUE*}, {ITEM.LOG.*}, {ITEM.VALUE*} {TRIGGER.ID}	
Polecenia parametrów użytkownika	Makra/referencje pozycyjne: $1...$9	
Scenariusz web		
Pola Nazwa, Agent, HTTP proxy	Definiowane przez użytkownika makra	
Pole Interwał aktualizacji	Definiowane przez użytkownika makra	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Pola Zmienne, Nagłówki, certyfikat SSL, plik klucza SSL	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Definiowane przez użytkownika makra	
Pola Użytkownik, Hasło, hasło klucza SSL	Definiowane przez użytkownika makra	
Krok scenariusza web *Pola Nazwa, Zmienne (tylko wartości)*	Definiowane przez użytkownika makra	
Krok scenariusza web *Pola Timeout, Wymagane kody statusu*	Definiowane przez użytkownika makra	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.

Lokalizacja	Obsługiwane makra	Komentarze
Krok scenariusza web *Pola URL, Nagłówki (nazwy i wartości), Wymagany ciąg*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Definiowane przez użytkownika makra	
Krok scenariusza web *Pole Post*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {HOST.PORT}	
Typ mediów web-hook		
Wartości parametrów	Makra Alert Wszystkie makra obsługiwane w powiadomieniach o problemach opartych na wyzwalaczu (patrz wyżej)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Skrypt JavaScript, nazwa parametru *Skrypt* JavaScript	Definiowane przez użytkownika makra (tylko globalne)	
Nazwa wpisu menu, URL wpisu menu	{EVENT.TAGS.<tag name>}	
Widżety		
Gauge widżet *Pole Opis*	Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Makra pozycja (z wyjątkiem {ITEM.STATE.ERROR}) Definiowane przez użytkownika makra (tylko globalne)	
Honeycomb widżet *Pole Etykieta główna/pomocnicza*		
Item value widżet *Pole Opis*		
Top hosts widżet *Kolumna Dane tekstowe*	Makra hosta: {HOST.CONN}, {HOST.DESCRIPTION}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Makra inwentarza hosta Definiowane przez użytkownika makra (tylko globalne)	
URL (Dynamiczny) widżet *Pole URL*	Makra hosta: {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.ID}, {HOST.IP}, {HOST.NAME}, {HOST.PORT} Definiowane przez użytkownika makra	
Kolumna **Nie-s-tandar-d-owy tekst***		
Monitoring > Problemy	Makra hosta (z wyjątkiem {HOST.METADATA}) Makra inwentarza hosta Makra pozycja (z wyjątkiem {ITEM.STATE.ERROR}) Makra wyzwalacza: {TRIGGER.ID} Makra zdarzenia: {EVENT.ID} Definiowane przez użytkownika makra	
Monitoring > Hosty	Makra hosta (z wyjątkiem {HOST.METADATA}) Makra inwentarza hosta Definiowane przez użytkownika makra	
Monitoring > Najnowsze dane	Makra hosta (z wyjątkiem {HOST.METADATA}) Makra inwentarza hosta Makra pozycja (z wyjątkiem {ITEM.STATE.ERROR}) Definiowane przez użytkownika makra	

Lokalizacja	Obsługiwane makra	Komentarze
Zbieranie danych > *Szablony*	Makra hosta: {HOST.DESCRIPTION}, {HOST.HOST}, {HOST.ID}, {HOST.NAME}	
Zbieranie danych > *Hosty*	Makra hosta (z wyjątkiem {HOST.METADATA}) Makra inwentarza hosta Definiowane przez użytkownika makra	
Inne lokalizacje		
Pole Czas pracy	Definiowane przez użytkownika makra (tylko globalne)	Pojedyncze makro musi wypełniać całe pole; wiele makr lub łączenie z tekstem nie jest obsługiwane.
Pole Timeouty dla typów pozycji (ustawienie globalne)		
Media użytkownika *Pole Gdy aktywne*	Definiowane przez użytkownika makra (tylko globalne)	
Typ mediów e-mail *pola Nazwa użytkownika, Hasło*	Definiowane przez użytkownika makra (tylko globalne)	Zalecane są tajne makra.
Zaplanowane raporty *pola Temat, Wiadomość*	Makra Data/godzina: {TIME}, {TIMESTAMP}	

Szczegóły makr

Ta lista zawiera szczegóły dotyczące każdego wbudowanego makra, pogrupowane według obszaru zastosowania:

- Makra akcji
- Makra alertów
- Makra daty/czasu
- Makra wykrywania
- Makra zdarzeń
  - Makra aktualizacji zdarzeń
  - Makra zdarzeń przyczyny/objawu
  - Makra zdarzeń odzyskiwania
- Makra eskalacji
- Makra funkcji
- Makra hosta
  - Makra hosta docelowego
- Makra grup hostów
- Makra inwentarza hosta
- Makra pozycji
- Makra powiadomień wykrywania niskiego poziomu
- Makra map
- Makra proxy
- Makra skryptów
- Makra usług
- Makra wyzwalaczy
- Makra nazw użytkowników

Makra akcji

{ACTION.ID}

Numeryczny identyfikator ID wyzwolonej akcji.

{ACTION.NAME}

Nazwa uruchomionej akcji.

Makra alertów

{ALERT.MESSAGE}

Wartość *Domyślna wiadomość* z konfiguracji akcji.

{ALERT.SENDTO}

Wartość *Send to* z konfiguracji mediów użytkownika.

{ALERT.SUBJECT}

Wartość *Default subject* z konfiguracji akcji.

Makra daty i czasu

{DATE}

Bieżąca data w formacie `yyyy.mm.dd`.

{TIME}

Bieżący czas w formacie `hh:mm:ss`.

{TIMESTAMP}

Bieżący czas w formacie znacznika czasu UNIX.

Makra wykrywania

{DISCOVERY.DEVICE.IPADDRESS}

Adres IP wykrytego urządzenia.
 Dostępne zawsze, nie zależy od dodania hosta.

{DISCOVERY.DEVICE.DNS}

Nazwa DNS wykrytego urządzenia.
 Zawsze dostępna, nie zależy od dodania hosta.

{DISCOVERY.DEVICE.STATUS}

Status wykrytego urządzenia (UP/DOWN).

{DISCOVERY.DEVICE.UPTIME}

Czas od ostatniej zmiany statusu wykrywania dla określonego urządzenia.
 Na przykład: 1h 29m 01s.
 Dla urządzeń ze statusem DOWN jest to okres ich niedostępności.

{DISCOVERY.RULE.NAME}

Nazwa reguły wykrywania, która wykryła obecność lub brak urządzenia/usługi.

{DISCOVERY.SERVICE.NAME}

Nazwa wykrytej usługi.
 Na przykład: HTTP.

{DISCOVERY.SERVICE.PORT}

Port wykrytej usługi.
 Na przykład: 80.

{DISCOVERY.SERVICE.STATUS}

Status wykrytej usługi (UP/DOWN).

{DISCOVERY.SERVICE.UPTIME}

Czas, jaki upłynął od ostatniej zmiany statusu wykrywania dla określonej usługi.
 Na przykład: 1h 29m 01s.
 Dla usług ze statusem DOWN jest to okres ich niedostępności.

Makra eskalacji

{ESC.HISTORY}

Dziennik wcześniej wysłanych powiadomień, ich krok eskalacji i status (*wysłane*, *w trakcie* lub *nieudane*).

Makra zdarzeń

{EVENT.ACK.STATUS}

Status potwierdzenia zdarzenia (Tak/Nie).

{EVENT.AGE}

Wiek zdarzenia, które wyzwoliło akcję, z dokładnością do jednej sekundy.
 Przydatne w eskalowanych wiadomościach. Na przykład: 9m 13s.

{EVENT.DATE}

Data zdarzenia, które wyzwoliło akcję, w formacie `rrrr.mm.dd`.
 Na przykład: 2025.04.14.

{EVENT.DURATION}

Czas trwania zdarzenia (różnica czasu między zdarzeniami problemu i odzyskania), z dokładnością do jednej sekundy.
 Przydatne w wiadomościach odzyskiwania problemu.

{EVENT.ID}

Numeryczny identyfikator zdarzenia, które wywołało akcję.

{EVENT.NAME}

Nazwa zdarzenia, które wyzwoliło akcję.

{EVENT.NSEVERITY}

Wartość liczbowa ważności zdarzenia.
 Możliwe wartości: *0* - Niesklasyfikowane, *1* - Informacja, *2* - Ostrzeżenie, *3* - Średni, *4* - Wysoki, *5* - Katastrofa.

{EVENT.OBJECT}

Wartość liczbowa obiektu zdarzenia.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Wykryty host, *2* - Wykryta usługa, *3* - Autorejestracja, *4* - Pozycja, *5* - Reguła wykrywania niskiego poziomu.

{EVENT.OPDATA}

Dane operacyjne bazowego wyzwalacza problemu.

{EVENT.SEVERITY}

Nazwa ważności zdarzenia.

{EVENT.SOURCE}

Wartość liczbowa źródła zdarzenia.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Wykrywanie, *2* - Autorejestracja, *3* - Wewnętrzne, *4* - Usługa.

{EVENT.STATUS}

Słowna wartość zdarzenia, które wyzwoliło akcję.

{EVENT.TAGS}

Lista tagów zdarzenia oddzielonych przecinkami.
Rozwijane do pustego ciągu znaków, jeśli nie istnieją żadne tagi.

{EVENT.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia.
Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.

{EVENT.TAGS.<tag name>}

Wartość tagu zdarzenia, do której odwołuje się nazwa tagu.
 Nazwa tagu zawierająca znaki inne niż alfanumeryczne (w tym wielobajtowe znaki UTF spoza języka angielskiego) musi być ujęta w podwójne cudzysłowy. Cudzysłowy i ukośniki odwrotne wewnątrz nazwy tagu ujętej w cudzysłowy muszą być poprzedzone ukośnikiem odwrotnym.

{EVENT.TIME}

Czas zdarzenia, które wyzwoliło akcję, w formacie `hh:mm:ss`.
 Na przykład: 12:57:53.

{EVENT.TIMESTAMP}

Znacznik czasu UNIX zdarzenia, które wyzwoliło akcję.

{EVENT.VALUE}

Wartość liczbowa zdarzenia, które wyzwoliło akcję.
 Możliwe wartości: *1* - problem, *0* - odzyskiwanie.

Makra zdarzeń przyczyny/objawu

{EVENT.CAUSE.ACK.STATUS}

Status potwierdzenia zdarzenia przyczyny (Tak/Nie).
 Używane w kontekście zdarzenia symptomu.

{EVENT.CAUSE.AGE}

Wiek zdarzenia będącego przyczyną, z dokładnością do jednej sekundy.
 Przydatne w eskalowanych wiadomościach.
 Używane w kontekście zdarzenia objawowego.

{EVENT.CAUSE.DATE}

Data zdarzenia będącego przyczyną w formacie `rrrr.mm.dd`.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.DURATION}

Czas trwania zdarzenia przyczyny (różnica czasu między zdarzeniami problemu i odzyskania), z dokładnością do jednej sekundy.
 Przydatne w wiadomościach odzyskiwania problemu.
 Używane w kontekście zdarzenia symptomu.

{EVENT.CAUSE.ID}

Numeryczny identyfikator zdarzenia będącego przyczyną.
 Używany w kontekście zdarzenia objawowego.

{EVENT.CAUSE.NAME}

Nazwa zdarzenia przyczyny.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.NSEVERITY}

Wartość liczbowa ważności zdarzenia będącego przyczyną.
 Możliwe wartości: *0* - Niesklasyfikowane, *1* - Informacja, *2* - Ostrzeżenie, *3* - Średnie, *4* - Wysokie, *5* - Katastrofa.
 Używane w kontekście zdarzenia objawowego.

{EVENT.CAUSE.OBJECT}

Wartość liczbowa obiektu zdarzenia będącego przyczyną.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Wykryty host, *2* - Wykryta usługa, *3* - Autorejestracja, *4* - Pozycja, *5* - Reguła wykrywania niskiego poziomu.
 Używane w kontekście zdarzenia będącego symptomem.

{EVENT.CAUSE.OPDATA}

Dane operacyjne bazowego wyzwalacza problemu.
 Używane w kontekście zdarzenia symptomu.

{EVENT.CAUSE.SEVERITY}

Nazwa poziomu ważności zdarzenia będącego przyczyną.
 Możliwe wartości: *Niesklasyfikowane*, *Informacja*, *Ostrzeżenie*, *Średni*, *Wysoki*, *Katastrofa*.
 Używane w kontekście zdarzenia będącego symptomem.

{EVENT.CAUSE.SOURCE}

Wartość liczbowa źródła zdarzenia przyczyny.
 Możliwe wartości: *0* - Wyzwalacz, *1* - Odkrywanie, *2* - Autorejestracja, *3* - Wewnętrzne.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.STATUS}

Słowna wartość zdarzenia będącego przyczyną.
 Używana w kontekście zdarzenia objawowego.

{EVENT.CAUSE.TAGS}

Lista tagów zdarzenia przyczyny, rozdzielonych przecinkami.
Rozwijane do pustego ciągu znaków, jeśli tagi nie istnieją.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia będącego przyczyną.
Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.
 Używana w kontekście zdarzenia objawowego.

{EVENT.CAUSE.TAGS.<tag name>}

Wartość tagu zdarzenia przyczyny, do której odwołuje się nazwa tagu.
 Nazwa tagu zawierająca znaki inne niż alfanumeryczne (w tym nieangielskie znaki wielobajtowe UTF) musi być ujęta w podwójne cudzysłowy. Cudzysłowy i ukośniki odwrotne wewnątrz nazwy tagu ujętej w cudzysłowy muszą być poprzedzone ukośnikiem odwrotnym.
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.TIME}

Czas zdarzenia będącego przyczyną w formacie `hh:mm:ss`.
 Używane w kontekście zdarzenia objawowego.

{EVENT.CAUSE.TIMESTAMP}

Znacznik czasu UNIX zdarzenia będącego przyczyną.
 Używany w kontekście zdarzenia objawowego.

{EVENT.CAUSE.UPDATE.HISTORY}

Dziennik aktualizacji problemu będącego przyczyną (potwierdzenia itp.).
 Używane w kontekście zdarzenia objawu.

{EVENT.CAUSE.VALUE}

Wartość liczbowa zdarzenia będącego przyczyną.
 Możliwe wartości: *1* - Problem, *0* - Odzyskiwanie.
 Używane w kontekście zdarzenia objawowego.

{EVENT.SYMPTOMS}

Lista zdarzeń objawów. Zawiera następujące szczegóły: nazwę hosta, nazwę zdarzenia, ważność, wiek, tagi usługi i wartości.
 To makro jest używane w kontekście zdarzenia przyczyny i zwraca informacje o zdarzeniach objawów.

Makra zdarzenia odzyskania

{EVENT.RECOVERY.DATE}

Data zdarzenia odzyskania w formacie `yyyy.mm.dd`.

{EVENT.RECOVERY.ID}

Numeryczny identyfikator zdarzenia odzyskania.

{EVENT.RECOVERY.NAME}

Nazwa zdarzenia odzyskiwania.

{EVENT.RECOVERY.STATUS}

Słowna wartość zdarzenia odzyskiwania.

{EVENT.RECOVERY.TAGS}

Lista tagów zdarzenia odzyskiwania rozdzielona przecinkami. Rozwijana do pustego ciągu, jeśli nie istnieją żadne tagi.

{EVENT.RECOVERY.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzenia odzyskiwania. Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.

{EVENT.RECOVERY.TIME}

Czas zdarzenia odzyskania w formacie `hh:mm:ss`.

{EVENT.RECOVERY.TIMESTAMP}

Znacznik czasu UNIX zdarzenia odzyskania.

{EVENT.RECOVERY.VALUE}

Wartość liczbowa zdarzenia odzyskiwania.

Makra aktualizacji zdarzenia

{EVENT.UPDATE.ACTION}

Czytelna dla człowieka nazwa akcji wykonanej podczas aktualizacji problemu.
 Przyjmuje następujące wartości: *acknowledged*, *unacknowledged*, *commented*, *changed severity from (original severity) to (updated severity)* oraz *closed* (w zależności od liczby akcji wykonanych podczas jednej aktualizacji).

{EVENT.UPDATE.ACTIONJSON}

Tablica JSON zawierająca szczegóły akcji wykonanej podczas aktualizacji problemu.
 Możliwe wartości właściwości JSON:
- true (dla właściwości `acknowledge`, `unacknowledge`, `close`, `unsuppress`, `cause` i `symptom`);
- <message string> (dla właściwości `message`);
- znacznik czasu (dla właściwości `suppress_until` i `timestamp`) lub 0 (dla właściwości `suppress_until`, jeśli wyciszenie jest bezterminowe);
- 0, 1, 2, 3, 4, 5 (dla właściwości `old` i `new` dotyczących severity).

 Na przykład: `{"acknowledge":true,"message":"Monthly maintenance.","severity":{"old":2,"new":1},"su`

{EVENT.UPDATE.DATE}

Data aktualizacji zdarzenia (potwierdzenia itp.) w formacie `rrrr.mm.dd`.

{EVENT.UPDATE.HISTORY}

Dziennik aktualizacji problemu (potwierdzeń itd.).

{EVENT.UPDATE.MESSAGE}

Wiadomość aktualizacji problemu.

{EVENT.UPDATE.NSEVERITY}

Wartość liczbowa nowego poziomu ważności zdarzenia ustawionego podczas operacji aktualizacji problemu.

{EVENT.UPDATE.SEVERITY}

Nazwa nowego poziomu ważności zdarzenia ustawionego podczas operacji aktualizacji problemu.

{EVENT.UPDATE.STATUS}

Wartość numeryczna statusu aktualizacji problemu.
 Możliwe wartości: *0* - webhook został wywołany z powodu zdarzenia problemu/odzyskania, *1* - operacja aktualizacji.

{EVENT.UPDATE.TIME}

Czas aktualizacji zdarzenia (potwierdzenia itp.) w formacie `hh:mm:ss`.

{EVENT.UPDATE.TIMESTAMP}

Znacznik czasu UNIX aktualizacji zdarzenia update (potwierdzenia itp.).

Makra funkcji

{FUNCTION.VALUE}

Wartość N-tej funkcji opartej na pozycji w wyrażeniu wyzwalacza w momencie wystąpienia zdarzenia.
 Uwzględniane są tylko funkcje, których pierwszym parametrem jest `/host/key`.

 Tego makra można używać z indeksem numerycznym, np. {FUNCTION.VALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. funkcję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{FUNCTION.RECOVERY.VALUE}

Wartość N-tej funkcji opartej na pozycji w wyrażeniu odzyskiwania w momencie wystąpienia zdarzenia.*
 Uwzględniane są tylko funkcje, których pierwszy parametr ma postać `/host/key`.

 Ta makro może być używana z indeksem numerycznym, np. {FUNCTION.RECOVERY.VALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. funkcję w wyrażeniu wyzwalacza. Zobacz indexed macros.

Makra hosta

{HOST.CONN}

Adres IP hosta lub nazwa DNS hosta, zależnie od ustawień hosta.
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach webowych makro zostanie rozwiązane do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwiązane do wartości 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.CONN<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{HOST.DESCRIPTION}

Opis hosta.

 To makro może być używane z indeks numerycznym jako {HOST.DESCRIPTION<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.DNS}

Nazwa DNS hosta.
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach WWW makro zostanie rozwiązane do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwiązane do wartości 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.DNS<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{HOST.HOST}

Techniczna nazwa hosta.
 Funkcje makr nie są obsługiwane dla tego makra, jeśli jest ono używane jako symbol zastępczy w pierwszym parametrze funkcji historii, na przykład `last(/{HOST.HOST}/{ITEM.KEY})`.

 To makro może być używane z indeksem numerycznym jako {HOST.HOST<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.ID}

ID hosta.

 To makro można użyć z indeksem numerycznym jako {HOST.ID<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.IP}

Adres IP hosta.
 W skryptach globalnych, polach IP/DNS interfejsu oraz scenariuszach webowych makro zostanie rozwiązane do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwiązane do wartości 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.IP<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{HOST.METADATA}

Metadane hosta.

 Używane tylko do aktywnej autorejestracji agenta.

{HOST.NAME}

Widoczna nazwa hosta.

 To makro może być używane z indeksem numerycznym jako {HOST.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{HOST.PORT}

Port hosta (agent).
 W globalnych skryptach, polach IP/DNS interfejsu oraz scenariuszach webowych makro zostanie rozwiązane do głównego interfejsu agent. Jeśli interfejs agent nie jest zdefiniowany, zostanie użyty główny interfejs SNMP. Jeśli interfejs SNMP również nie jest zdefiniowany, zostanie użyty główny interfejs JMX. Jeśli interfejs JMX także nie jest zdefiniowany, zostanie użyty główny interfejs IPMI. Jeśli host nie ma żadnego interfejsu, makro zostanie rozwiązane do wartości 'UNKNOWN'.

 To makro może być używane z indeksem numerycznym jako {HOST.PORT<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

Makra hosta docelowego

{HOST.TARGET.CONN}

Adres IP lub nazwa DNS hosta docelowego.

{HOST.TARGET.DNS}

Nazwa DNS docelowego hosta.

{HOST.TARGET.HOST}

Techniczna nazwa docelowego hosta.

{HOST.TARGET.IP}

Adres IP docelowego hosta.

{HOST.TARGET.NAME}

Widoczna nazwa docelowego hosta.

Makra grup hostów

{HOSTGROUP.ID}

ID grupy hostów.

Makra inwentarza hosta

{INVENTORY.ALIAS}

Pole *Alias* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.ALIAS<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.ASSET.TAG}

Pole *Asset tag* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.ASSET.TAG<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.CHASSIS}

Pole *Chassis* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.CHASSIS<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.CONTACT}

Pole *Contact* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.CONTACT<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.CONTRACT.NUMBER}

Pole *Contract number* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.CONTRACT.NUMBER<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.DEPLOYMENT.STATUS}

Pole *Deployment status* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.DEPLOYMENT.STATUS<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.HARDWARE}

Pole *Hardware* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HARDWARE<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HARDWARE.FULL}

Pole *Sprzęt (pełne szczegóły)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HARDWARE.FULL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HOST.NETMASK}

Pole *maska podsieci hosta* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.HOST.NETMASK<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HOST.NETWORKS}

Pole *Host networks* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HOST.NETWORKS<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.HOST.ROUTER}

Pole *Host router* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.HOST.ROUTER<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.HW.ARCH}

Pole *Hardware architecture* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.HW.ARCH<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.HW.DATE.DECOMM}

Pole *Date hardware decommissioned* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.HW.DATE.DECOMM<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.HW.DATE.EXPIRY}

Pole *Data wygaśnięcia konserwacji sprzętu* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.HW.DATE.EXPIRY<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.HW.DATE.INSTALL}

Pole *Date hardware installed* w inwentarzu hosta.

 Tego makra można używać z indeksami numerycznymi, np. {INVENTORY.HW.DATE.INSTALL<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.HW.DATE.PURCHASE}

Pole *Date hardware purchased* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.HW.DATE.PURCHASE<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.INSTALLER.NAME}

Pole *Nazwa instalatora* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.INSTALLER.NAME<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.LOCATION}

Pole *Location* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.LOCATION<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.LOCATION.LAT}

Pole *Location latitude* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.LOCATION.LAT<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.LOCATION.LON}

Pole *Długość geograficzna lokalizacji* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.LOCATION.LON<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.MACADDRESS.A}

Pole *MAC address A* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.MACADDRESS.A<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.MACADDRESS.B}

Pole *MAC address B* w inwentarzu hosta.

 Ta makro może być używana z indeksami numerycznymi, np. {INVENTORY.MACADDRESS.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.MODEL}

Pole *Model* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.MODEL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.NAME}

Pole *Name* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.NOTES}

Pole *Notes* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.OOB.IP}

Pole *adres IP OOB* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OOB.IP<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OOB.NETMASK}

Pole *maska podsieci OOB* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.OOB.NETMASK<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OOB.ROUTER}

Pole *OOB router* w inwentarzu hosta.

 Tego makra można używać z indeksami numerycznymi, np. {INVENTORY.OOB.ROUTER<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.OS}

Pole *OS* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.OS<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OS.FULL}

Pole *OS (szczegółowe informacje)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.OS.FULL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.OS.SHORT}

Pole *OS (Short)* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.OS.SHORT<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.CELL}

Pole *Primary POC cell* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.CELL<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.EMAIL}

Pole *Primary POC email* w inwentarzu hosta.

 Ta makro może być używana z indeks numerycznym, np. {INVENTORY.POC.PRIMARY.EMAIL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.NAME}

Pole *Primary POC name* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.NOTES}

Pole *Primary POC notes* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.PHONE.A}

Pole *Primary POC phone A* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.PHONE.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.PHONE.B}

Pole *Primary POC phone B* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.PHONE.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.PRIMARY.SCREEN}

Pole *Primary POC screen name* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.POC.PRIMARY.SCREEN<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.SECONDARY.CELL}

Pole *Secondary POC cell* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.CELL<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.SECONDARY.EMAIL}

Pole *Secondary POC email* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.POC.SECONDARY.EMAIL<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.SECONDARY.NAME}

Pole *Secondary POC name* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.POC.SECONDARY.NAME<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.SECONDARY.NOTES}

Pole *Secondary POC notes* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.POC.SECONDARY.PHONE.A}

Pole *Secondary POC phone A* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.PHONE.A<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.PHONE.B}

Pole *Secondary POC phone B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.PHONE.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.POC.SECONDARY.SCREEN}

Pole *Nazwa ekranu drugiej osoby kontaktowej* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.POC.SECONDARY.SCREEN<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SERIALNO.A}

Pole *Serial number A* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SERIALNO.A<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SERIALNO.B}

Pole *Serial number B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SERIALNO.B<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SITE.ADDRESS.A}

Pole *Site address A* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.SITE.ADDRESS.A<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SITE.ADDRESS.B}

Pole *Site address B* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.ADDRESS.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.ADDRESS.C}

Pole *Site address C* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.ADDRESS.C<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.CITY}

Pole *Miasto lokalizacji* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.CITY<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.COUNTRY}

Pole *Kraj lokalizacji* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.SITE.COUNTRY<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.NOTES}

Pole *Site notes* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.SITE.NOTES<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SITE.RACK}

Pole *Site rack location* w inwentarzu hosta.

 To makro może być używane z indeks numerycznym, np. {INVENTORY.SITE.RACK<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SITE.STATE}

Pole *Stan województwa/prowincji* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.SITE.STATE<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SITE.ZIP}

Pole *Site ZIP/postal* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.SITE.ZIP<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE}

Pole *Software* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.SOFTWARE<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.A}

Pole *Software application A* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.A<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SOFTWARE.APP.B}

Pole *Software application B* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.SOFTWARE.APP.B<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.C}

Pole *Software application C* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.C<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SOFTWARE.APP.D}

Pole *Software application D* w inwentarzu hosta.

 Tego makra można używać z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.D<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.SOFTWARE.APP.E}

Pole *Software application E* w inwentarzu hosta.

 Ta makro może być używana z indeksem numerycznym, np. {INVENTORY.SOFTWARE.APP.E<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz indexed macros.

{INVENTORY.SOFTWARE.FULL}

Pole *Oprogramowanie (pełne szczegóły)* w inwentarzu hosta.

 Ta makro może być używana z indeks numerycznym, np. {INVENTORY.SOFTWARE.FULL<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.TAG}

Pole *Tag* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.TAG<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.TYPE}

Pole *Type* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.TYPE<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.TYPE.FULL}

Pole *Typ (pełne szczegóły)* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.TYPE.FULL<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.URL.A}

Pole *URL A* w inwentarzu hosta.

 Ta makro może być używana z indeks numerycznym, np. {INVENTORY.URL.A<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.URL.B}

Pole *URL B* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.URL.B<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.URL.C}

Pole *URL C* w inwentarzu hosta.

 To makro może być używane z indeksami numerycznymi, np. {INVENTORY.URL.C<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{INVENTORY.VENDOR}

Pole *Vendor* w inwentarzu hosta.

 To makro może być używane z indeksem numerycznym, np. {INVENTORY.VENDOR<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra pozycji

{ITEM.DESCRIPTION}

Opis N-tej pozycji w wyrażeniu wyzwalacza, które spowodowało powiadomienie.

 Tego makra można używać z indeksem numerycznym jako {ITEM.DESCRIPTION<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.DESCRIPTION.ORIG}

Oryginalny opis (z nierozwiązanymi makrami) N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.DESCRIPTION.ORIG<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.ID}

Numeryczny identyfikator N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.ID<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz indexed macros.

{ITEM.KEY}

Klucz N-tej pozycji w wyrażeniu wyzwalacza, które spowodowało powiadomienie.
 Funkcje makr nie są obsługiwane dla tego makra, jeśli jest ono używane jako symbol zastępczy w pierwszym parametrze funkcji historii, na przykład `last(/{HOST.HOST}/{ITEM.KEY}`

To makro może być używane z indeksem numerycznym jako {ITEM.KEY<1-9>}, aby wskazać pierwszy, drugi, trzeci itd. host w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.KEY.ORIG}

Oryginalny klucz (z nierozwiniętymi makrami) N-tej pozycji w wyrażeniu wyzwalacza, która spowodowała powiadomienie.

 To makro może być używane z indeksem numerycznym jako {ITEM.KEY.ORIG<1-9>}, aby wskazać pierwszego, drugiego, trzeciego itd. hosta w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE}

Najnowsza wartość N-tej pozycji w wyrażeniu wyzwalacza, które spowodowało powiadomienie.
 Zostanie rozwiązana do *UNKNOWN* w frontendzie, jeśli najnowsza wartość historii została zebrana ponad czas *Maksymalnego okresu wyświetlania historii* temu (ustawiane w sekcji menu *Administracja > Ogólne*).
 Gdy jest używana w nazwie problemu, makro nie zostanie rozwiązane do najnowszej wartości pozycji podczas przeglądania zdarzeń problemu; zamiast tego zachowa wartość pozycji z momentu wystąpienia problemu.
Gdy jest używana w powiadomieniach, w niektórych przypadkach makro może nie zostać rozwiązane do najnowszej wartości pozycji w momencie wyzwolenia wyzwalacza. Na przykład, jeśli pozycja szybko otrzyma dwie wartości, ”A” i ”B”, a wyzwalacz uruchomi się dla ”A”, powiadomienia mogą pokazać ”B” jako najnowszą wartość z powodu niewielkiego opóźnienia przetwarzania - najnowsza wartość pozycji zmieniła się między momentem uruchomienia wyzwalacza a utworzeniem powiadomienia. Aby tego uniknąć, można użyć makra {ITEM.VALUE}, które rozwiązuje się do wartości z momentu wyzwolenia wyzwalacza, zapewniając użycie poprawnej wartości w powiadomieniu.
 Jest to alias `last(/{HOST.HOST}/{ITEM.KEY})`.
 Rozwiązana wartość dla pozycji tekstowych/logów jest skracana do 20 znaków przez frontend w następujących miejscach:
- Dane operacyjne;
- Opis wyzwalacza;
- URL-e wyzwalacza;
- Etykiety URL-i wyzwalacza;
- Opis widżetu wartości pozycji.
Aby uzyskać pełną wartość, można użyć funkcji makr, ponieważ serwer nie skraca żadnych wartości. Na przykład: `{{ITEM.LASTVALUE}.regsub("(.*)", \1)}`.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.AGE}

Czas, jaki upłynął między ostatnim pobraniem wartości pozycji a oceną makra.
 Wyświetlany w formacie czytelnym dla człowieka (np. 1m 45s).

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.AGE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.DATE}

Data, kiedy została pobrana najnowsza wartość pozycji.
 Wyświetlane w formacie `YYYYMMDD`.
 W nazwach wyzwalaczy (na liście *Monitoring* > *Problemy*), nazwach zdarzeń oraz nazwach i wartościach tagów używana jest strefa czasowa serwera. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

 Ta makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.DATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.TIME}

Czas, w którym została pobrana najnowsza wartość pozycji.
 Wyświetlany w formacie `HHMMSS`.
 W nazwach wyzwalaczy (na liście *Monitoring* > *Problemy*), nazwach zdarzeń oraz nazwach i wartościach tagów używana jest strefa czasowa serwera. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

 Ta makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.TIME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LASTVALUE.TIMESTAMP}

Znacznik czasu UNIX, kiedy została pobrana najnowsza wartość pozycji.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LASTVALUE.TIMESTAMP<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.AGE}

Wiek zdarzenia z dziennika.
Z dokładnością do sekundy.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.AGE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.DATE}

Data, w której wpis dziennika został zapisany do dziennika.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.DATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.EVENTID}

ID zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń systemu Windows.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.EVENTID<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.NSEVERITY}

Numeryczny poziom ważności zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń systemu Windows.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.NSEVERITY<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.SEVERITY}

Słowny poziom ważności zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń systemu Windows.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.SEVERITY<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.LOG.SOURCE}

Źródło zdarzenia w dzienniku zdarzeń.
 Tylko do monitorowania dziennika zdarzeń systemu Windows.

 Ta makro może być używana z indeksem numerycznym, np. {ITEM.LOG.SOURCE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{ITEM.LOG.TIME}

Czas, w którym wpis dziennika został zapisany w dzienniku.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.TIME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{ITEM.LOG.TIMESTAMP}

Znacznik czasu UNIX, kiedy wpis dziennika został zapisany w dzienniku.

 To makro może być używane z indeksem numerycznym, np. {ITEM.LOG.TIMESTAMP<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.NAME}

Nazwa pozycji ze wszystkimi rozwiniętymi makrami.

 To makro może być używane z indeksem numerycznym, np. {ITEM.NAME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.NAME.ORIG}

Oryginalna nazwa pozycja (z nierozwiniętymi makrami).

 To makro może być używane z indeksem numerycznym, np. {ITEM.NAME.ORIG<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.STATE}

Najnowszy stan N-tej pozycji w wyrażeniu wyzwalacza, które spowodowało powiadomienie.
 Możliwe wartości: *Not supported*, *Normal*.

 To makro może być używane z indeksem numerycznym, np. {ITEM.STATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{ITEM.STATE.ERROR}

Komunikat błędu zawierający szczegóły, dlaczego pozycja stała się nieobsługiwana.
 Jeśli pozycja przejdzie w stan nieobsługiwany, a następnie natychmiast ponownie stanie się obsługiwana, pole błędu może być puste.

{ITEM.VALUE}

Rozwiązuje się do jednej z dwóch wartości:

- Historycznej wartości N-tej pozycji w wyrażeniu wyzwalacza, z chwili wystąpienia zdarzenia, jeśli jest używana w kontekście zmiany stanu wyzwalacza, na przykład podczas wyświetlania zdarzeń lub wysyłania powiadomień;
- Najnowszej wartości N-tej pozycji w wyrażeniu wyzwalacza, jeśli jest używana bez kontekstu zmiany stanu wyzwalacza, na przykład podczas wyświetlania listy wyzwalaczy w oknie wyboru typu pop-up. W tym przypadku działa tak samo jak {ITEM.LASTVALUE}.

W pierwszym przypadku zostanie rozwiązana do *UNKNOWN*, jeśli wartość historyczna została już usunięta lub nigdy nie została zapisana.
W drugim przypadku, i tylko we frontendzie, zostanie rozwiązana do *UNKNOWN*, jeśli najnowsza wartość historyczna została zebrana ponad *Max history display period* temu (ustawiane w menu *Administration > General*).
Wartość rozwiązana dla pozycji tekstowych/logów jest przycinana do 20 znaków przez frontend w następujących miejscach:
- Operational data;
- Trigger description;
- Trigger URLs;
- Trigger URL labels;
- Description of the item value widget.
Aby uzyskać pełną wartość, można użyć funkcji makr, ponieważ serwer nie przycina żadnych wartości. Na przykład: `{{ITEM.VALUE}.regsub("(.*)", \1)}`.

To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUE.AGE}

Czas, który upłynął między pobraniem wartości pozycja a wykonaniem makra.
 Wyświetlany w formacie czytelnym dla człowieka (np. 1m 45s).

 To makro może być używane z indeksem numerycznym, np. {ITEM.VALUE.AGE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{ITEM.VALUE.DATE}

Data, kiedy została pobrana wartość pozycji.
 Wyświetlana w formacie `YYYYMMDD`.
 Strefa czasowa serwera jest używana w nazwach wyzwalaczy (na liście *Monitoring* > *Problemy*), nazwach zdarzeń oraz nazwach i wartościach tagów. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

Ta makro może być używane z indeksem numerycznym, np. {ITEM.VALUE.DATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{ITEM.VALUE.TIME}

Czas, w którym została zebrana wartość pozycji.
 Wyświetlany w formacie `HHMMSS`.
 Strefa czasowa serwera jest używana w nazwach wyzwalaczy (na liście *Monitoring* > *Problemy*), nazwach zdarzeń oraz nazwach i wartościach tagów. We wszystkich pozostałych przypadkach używana jest strefa czasowa użytkownika.

Ten makro może być używany z indeksem numerycznym, np. {ITEM.VALUE.TIME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUE.TIMESTAMP}

Znacznik czasu UNIX, w którym została pobrana wartość pozycji.

 To makro może być używane z indeksami numerycznymi, np. {ITEM.VALUE.TIMESTAMP<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{ITEM.VALUETYPE}

Typ wartości N-tej pozycji w wyzwalaczu, który spowodował powiadomienie.
 Możliwe wartości: *0* - Liczba zmiennoprzecinkowa, *1* - Znak, *2* - Dziennik, *3* - Liczba całkowita bez znaku, *4* - Tekst.

 Ta makro może być używana z indeksem numerycznym, np. {ITEM.VALUETYPE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyzwalaczu. Zobacz indexed macros.

Makra powiadomień wykrywania niskiego poziomu

{LLDRULE.DESCRIPTION}

Opis reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.DESCRIPTION.ORIG}

Oryginalny opis (z nierozwiniętymi makrami) reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.ID}

Numeryczny identyfikator reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.KEY}

Klucz reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.KEY.ORIG}

Oryginalny klucz (z nierozwiniętymi makrami) reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.NAME}

Nazwa reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.NAME.ORIG}

Oryginalna nazwa (z nierozwiniętymi makrami) reguły wykrywania niskiego poziomu, która spowodowała powiadomienie.

{LLDRULE.STATE}

Najnowszy stan reguły wykrywania niskiego poziomu.
 Możliwe wartości: *Not supported*, *Normal*.

 To makro może być używane z indeksem numerycznym, np. {LLDRULE.STATE<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz indexed macros.

{LLDRULE.STATE.ERROR}

Komunikat błędu zawierający szczegóły, dlaczego reguła wykrywania niskiego poziomu stała się nieobsługiwana.
 Jeśli reguła wykrywania niskiego poziomu przejdzie w stan nieobsługiwany, a następnie natychmiast ponownie stanie się obsługiwana, pole błędu może być puste.

Makra mapy

{MAP.ID}

ID mapy sieciowej.

{MAP.NAME}

Nazwa mapy sieciowej.

Makra proxy

{PROXY.DESCRIPTION}

Opis proxy.
 Rozwiązuje się do jednego z poniższych:

- Proxy n-tej pozycji w wyrażeniu wyzwalacza (w powiadomieniach opartych na wyzwalaczu). Możesz tutaj używać makr indeksowanych;
- Proxy, które wykonało wykrywanie (w powiadomieniach o wykrywaniu). Użyj tutaj {PROXY.DESCRIPTION}, bez indeksowania;
- Proxy, do którego zarejestrował się aktywny agent (w powiadomieniach o autorejestracji). Użyj tutaj {PROXY.DESCRIPTION}, bez indeksowania.

To makro może być używane z indeksem numerycznym, np. {PROXY.DESCRIPTION<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

{PROXY.NAME}

Nazwa proxy.
 Rozwiązuje się do jednego z poniższych:

- Proxy n-tej pozycji w wyrażeniu wyzwalacza (w powiadomieniach opartych na wyzwalaczach). Możesz tutaj używać makr indeksowanych;
- Proxy, które wykonało wykrywanie (w powiadomieniach o wykrywaniu). Użyj tutaj {PROXY.NAME}, bez indeksowania;
- Proxy, do którego zarejestrował się aktywny agent (w powiadomieniach o autorejestracji). Użyj tutaj {PROXY.NAME}, bez indeksowania.

To makro może być używane z indeksem numerycznym, np. {PROXY.NAME<1-9>}, aby wskazać pierwszą, drugą, trzecią itd. pozycję w wyrażeniu wyzwalacza. Zobacz makra indeksowane.

Makra skryptu

{MANUALINPUT}

Wartość ręcznego wejścia określona przez użytkownika w czasie wykonywania skryptu.

Makra usługi

{SERVICE.DESCRIPTION}

Opis usługi z rozwiązanymi makrami.

{SERVICE.ID}

Numeryczny identyfikator usługi, która wywołała akcję.

{SERVICE.NAME}

Nazwa usługi z rozwiniętymi makrami.

{SERVICE.ROOTCAUSE}

Lista zdarzeń problemowych wyzwalaczy, które spowodowały awarię usługi, posortowana według ważności i nazwy hosta.
 Zawiera następujące szczegóły: nazwa hosta, nazwa zdarzenia, ważność, wiek, tagi usługi i wartości.

{SERVICE.TAGS}

Lista tagów zdarzeń usługi rozdzielonych przecinkami.
 Tagi zdarzeń usługi można zdefiniować w sekcji tagów konfiguracji usługi. Rozwijane do pustego ciągu, jeśli nie istnieją żadne tagi.

{SERVICE.TAGSJSON}

Tablica JSON zawierająca obiekty tagów zdarzeń usługi.
 Tagi zdarzeń usługi można zdefiniować w sekcji tagów konfiguracji usługi. Rozwijana do pustej tablicy, jeśli nie istnieją żadne tagi.

{SERVICE.TAGS.<tag name>}

Wartość tagu zdarzenia usługi, do której odwołuje się nazwa tagu.
 Tagi zdarzenia usługi można zdefiniować w sekcji tagów konfiguracji usługi.
 Nazwa tagu zawierająca znaki inne niż alfanumeryczne (w tym wielobajtowe znaki UTF spoza języka angielskiego) musi być ujęta w podwójne cudzysłowy. Cudzysłowy i ukośniki odwrotne wewnątrz nazwy tagu ujętej w cudzysłowy muszą być poprzedzone ukośnikiem odwrotnym.

Makra wyzwalaczy

{TRIGGER.DESCRIPTION}

Opis wyzwalacza.
 Wszystkie makra obsługiwane w opisie wyzwalacza zostaną rozwinięte, jeśli `{TRIGGER.DESCRIPTION}` zostanie użyte w tekście powiadomienia.

{TRIGGER.EVENTS.ACK}

Liczba potwierdzonych zdarzeń dla elementu mapy na mapach lub dla wyzwalacza, który wygenerował bieżące zdarzenie w powiadomieniach.

{TRIGGER.EVENTS.PROBLEM.ACK}

Liczba potwierdzonych zdarzeń problemów dla wszystkich wyzwalaczy, niezależnie od ich stanu.

{TRIGGER.EVENTS.PROBLEM.UNACK}

Liczba niepotwierdzonych zdarzeń problemowych dla wszystkich wyzwalaczy, niezależnie od ich stanu.

{TRIGGER.EVENTS.UNACK}

Liczba niepotwierdzonych zdarzeń dla elementu mapy na mapach lub dla wyzwalacza, który wygenerował bieżące zdarzenie w powiadomieniach.

{TRIGGER.EXPRESSION}

Wyrażenie wyzwalacza.

{TRIGGER.EXPRESSION.EXPLAIN}

Częściowo obliczone wyrażenie wyzwalacza.
 Funkcje oparte na pozycjach są obliczane i zastępowane wynikami w momencie generowania zdarzenia, natomiast wszystkie pozostałe funkcje są wyświetlane tak, jak zostały zapisane w wyrażeniu. Może być używane do debugowania wyrażeń wyzwalacza.

{TRIGGER.EXPRESSION.RECOVERY}

Wyrażenie odzyskiwania wyzwalacza, jeśli *OK event generation* w konfiguracji wyzwalacza jest ustawione na „Recovery expression”; w przeciwnym razie zwracany jest pusty ciąg.

{TRIGGER.EXPRESSION.RECOVERY.EXPLAIN}

Częściowo obliczone wyrażenie odzyskiwania wyzwalacza.
 Funkcje oparte na pozycjach są obliczane i zastępowane wynikami w momencie generowania zdarzenia, natomiast wszystkie pozostałe funkcje są wyświetlane tak, jak zostały zapisane w wyrażeniu. Może być używane do debugowania wyrażeń odzyskiwania wyzwalacza.

{TRIGGER.HOSTGROUP.NAME}

Posortowana (według zapytania SQL) lista grup hostów, rozdzielonych przecinkiem i spacją, w których zdefiniowany jest wyzwalacz.

{TRIGGER.ID}

Numeryczny identyfikator wyzwalacza, który uruchomił tę akcję.

{TRIGGER.NAME}

Nazwa wyzwalacza z rozwiniętymi makrami.
 Zwróć uwagę, że `{EVENT.NAME}` może być używane w akcjach do wyświetlania nazwy wywołanego zdarzenia/problemu z rozwiniętymi makrami.

{TRIGGER.NAME.ORIG}

Oryginalna nazwa wyzwalacza (z nierozwiniętymi makrami).

{TRIGGER.NSEVERITY}

Numeryczny poziom ważności wyzwalacza.
 Możliwe wartości: *0* - Niesklasyfikowany, *1* - Informacja, *2* - Ostrzeżenie, *3* - Średni, *4* - Wysoki, *5* - Katastrofa.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.ACK}

Liczba potwierdzonych zdarzeń problemów dla wyzwalaczy w stanie problemu.

{TRIGGER.PROBLEM.EVENTS.PROBLEM.UNACK}

Liczba niepotwierdzonych zdarzeń problemów dla wyzwalaczy w stanie problemu.

{TRIGGER.SEVERITY}

Nazwa ważności wyzwalacza.
 Może być zdefiniowana w *Administration* > *General* > *Trigger displaying options*.

{TRIGGER.STATE}

Najnowszy stan wyrażenia wyzwalacza.
 Możliwe wartości: *Nieznany*, *Normalny*.

{TRIGGER.STATE.ERROR}

Komunikat błędu zawierający szczegóły, dlaczego wyzwalacz stał się nieobsługiwany.
 Jeśli wyzwalacz przejdzie w stan nieobsługiwany, a następnie natychmiast ponownie stanie się obsługiwany, pole błędu może być puste.

{TRIGGER.STATUS}

Wartość wyzwalacza w momencie wykonania kroku operacji.
 Możliwe wartości: *Problem*, *OK*.

{TRIGGER.TEMPLATE.NAME}

Posortowana (według zapytania SQL) lista szablonów, oddzielona przecinkami i spacjami, w których zdefiniowano wyzwalacz, lub *UNKNOWN*, jeśli wyzwalacz jest zdefiniowany w hoście.

{TRIGGER.URL}

URL wyzwalacza.

{TRIGGER.URL.NAME}

Etykieta dla URL wyzwalacza.

{TRIGGER.VALUE}

Bieżąca numeryczna wartość wyzwalacza.
 Możliwe wartości: *0* - wyzwalacz jest w stanie OK, *1* - wyzwalacz jest w stanie Problem.

{TRIGGERS.ACK}

Liczba potwierdzonych wyzwalaczy dla elementu mapy, bez uwzględniania stanu wyzwalacza.
 Wyzwalacz jest uznawany za potwierdzony, jeśli wszystkie jego zdarzenia problemowe są potwierdzone.

{TRIGGERS.PROBLEM.ACK}

Liczba potwierdzonych wyzwalaczy problemów dla elementu mapy.
 Wyzwalacz jest uznawany za potwierdzony, jeśli wszystkie jego zdarzenia problemowe są potwierdzone.

{TRIGGERS.PROBLEM.UNACK}

Liczba niepotwierdzonych wyzwalaczy problemów dla elementu mapy.
 Wyzwalacz jest uznawany za niepotwierdzony, jeśli co najmniej jedno z jego zdarzeń problemu nie zostało potwierdzone.

{TRIGGERS.UNACK}

Liczba niepotwierdzonych wyzwalaczy dla elementu mapy, niezależnie od stanu wyzwalacza.
 Wyzwalacz jest uznawany za niepotwierdzony, jeśli co najmniej jedno z jego zdarzeń problemowych nie zostało potwierdzone.

Makra nazwy użytkownika

{USER.FULLNAME}

Imię, nazwisko i nazwa użytkownika użytkownika, który dodał potwierdzenie zdarzenia lub uruchomił skrypt.

{USER.NAME}

Nazwa użytkownika, który uruchomił skrypt.

{USER.SURNAME}

Nazwisko użytkownika, który uruchomił skrypt.

{USER.USERNAME}

Nazwa użytkownika, który uruchomił skrypt.

Makra indeksowane

Składnia makra indeksowanego **{MACRO<1-9>}** może odwoływać się wyłącznie do N-tej pozycji, funkcji lub hosta w polu *Expression* wyzwalacza:

- **{HOST.IP1}, {HOST.IP2}, {HOST.IP3}** zwracają adresy IP pierwszego, drugiego i trzeciego hosta w wyrażeniu wyzwalacza (jeśli występują).
- **{ITEM.VALUE1}, {ITEM.VALUE2}, {ITEM.VALUE3}** zwracają wartości pierwszej, drugiej i trzeciej pozycji w wyrażeniu wyzwalacza w momencie wystąpienia zdarzenia (jeśli występują).
- **{FUNCTION.VALUE1}, {FUNCTION.VALUE2}, {FUNCTION.VALUE3}** zwracają wartości pierwszej, drugiej i trzeciej funkcji opartej na pozycji w momencie wystąpienia zdarzenia (jeśli występują).

W kontekście wyzwalaczy makra indeksowane zawsze odnoszą się do pola *Expression* konfiguracji wyzwalacza, a nie do *Recovery expression*. Na przykład w zdarzeniu odzyskiwania **{ITEM.VALUE2}** zwróci wartość drugiej pozycji z wyrażenia problemu w momencie odzyskiwania.

Makro **{HOST.HOST<1-9>}** jest również obsługiwane w makrze wyrażenia `{?func(/host/key,param)}` w **nazwach wykresów**. Na przykład `{?func(/{HOST.HOST2}/key,param)}` w nazwie wykresu zostanie rozwiązane do hosta drugiej pozycji na wykresie.

Warning:
Makra indeksowane nie zostaną rozwiązane w żadnym innym kontekście poza przypadkami wymienionymi tutaj. W innych kontekstach należy używać makr bez indeksu (**{HOST.HOST}, {HOST.IP}** itd.).

Makra użytkownika obsługiwane przez lokalizację

Przegląd

Ta sekcja zawiera listę miejsc, w których obsługiwane są makra definiowane przez użytkownika.

Note:
Dla *Akcji*, *Odkrywania sieci*, *Proxy* oraz wszystkich lokalizacji wymienionych w sekcji *Inne lokalizacje* na tej stronie obsługiwane są tylko makra użytkownika na poziomie globalnym. W wymienionych lokalizacjach makra na poziomie hosta i na poziomie szablonu nie będą rozwijane.

Note:
Aby dostosować wartości makr (na przykład skrócić je lub wyodrębnić określone podciągi), można użyć funkcji makr.

Akcje

W akcjach makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Powiadomienia i polecenia oparte na wyzwalaczach		tak
Wewnętrzne powiadomienia oparte na wyzwalaczach		tak
Powiadomienia o aktualizacji problemu		tak
Powiadomienia i polecenia oparte na usługach		tak
Powiadomienia o aktualizacji usługi		tak
Warunek okresu czasu		nie
Operacje		
	Domyślny czas trwania kroku operacji	nie
	Czas trwania kroku	nie

Hosty/prototypy hostów

W konfiguracji hosta i prototypu hosta makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
IP/DNS interfejsu		tylko DNS
Port interfejsu		nie
SNMP v1, v2		
	Społeczność SNMP	tak
SNMP v3		
	Nazwa kontekstu	tak
	Nazwa bezpieczeństwa	tak
	Hasło uwierzytelniania	tak
	Hasło prywatności	tak
IPMI		
	Nazwa użytkownika	tak
	Hasło	tak
Tagi[2]		
	Nazwy tagów	tak

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
	Wartości tagów	tak

Pozycje / prototypy pozycji

W konfiguracji pozycji lub prototypu pozycji makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Nazwa pozycji	tak
Parametry klucza pozycji	tak[4]
Interwał aktualizacji	nie
Interwały niestandardowe	nie
Limit czasu (dostępny dla obsługiwanych typów pozycji)	nie
Przechowuj przez (dla historii i trendów)	nie
Opis	tak
Pozycja obliczana/agregująca	
Formuła (stałe wyrażeń i parametry funkcji; parametry klucza pozycji; (*tylko pozycja agregująca*) warunki filtrowania (nazwa grupy hostów i nazwa znacznika))	tak
Monitor bazy danych	
Nazwa użytkownika	tak
Hasło	tak
Zapytanie SQL	tak
agent HTTP	
URL[3]	tak
Pola zapytania	tak
Treść żądania	tak

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Nagłówki (nazwy i wartości)	tak
Wymagane kody stanu	tak
proxy HTTP	tak
Nazwa użytkownika uwierzytelniania HTTP	tak
Hasło uwierzytelniania HTTP	tak
Plik certyfikatu SSl	tak
Plik klucza SSl	tak
Hasło klucza SSl	tak
Dozwolone hosty	tak
agent JMX	
Punkt końcowy JMX	tak
Pozycja skryptowa	
Nazwy i wartości parametrów	tak
Pozycja przeglądarki	
Nazwy i wartości parametrów	tak
Pozycja zapytania telemetrycznego	
Przesunięcie czasowe	nie
Limit zakresu wstecz	nie
Szczegółowość	nie
agent SNMP	
OID SNMP	tak
agent SSH	
Nazwa użytkownika	tak
Plik klucza publicznego	tak
Plik klucza prywatnego	tak
Hasło	tak
Skrypt	tak
agent TELNET	
Nazwa użytkownika	tak
Hasło	tak
Skrypt	tak
Trapper Zabbix	
Dozwolone hosty	tak
Znaczniki[2]	
Nazwy znaczników	tak
Wartości znaczników	tak
Kroki wstępnego przetwarzania	
Parametry (w tym skrypty niestandardowe)	tak
Niestandardowe parametry obsługi błędów (pola *Ustaw wartość* i *Ustaw błąd na*)	tak

Odkrywanie niskiego poziomu

W regule odkrywania niskiego poziomu makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/
Parametry klucza		tak
Interwał aktualizacji		nie
Interwał niestandardowy		nie
Limit czasu (dostępny dla obsługiwanych typów pozycji)		nie
Usuń utracone zasoby		nie
Wyłącz utracone zasoby		nie
Opis		tak
SNMP agent		
	OID SNMP	tak
SSH agent		
	Nazwa użytkownika	tak
	Plik klucza publicznego	tak
	Plik klucza prywatnego	tak
	Hasło	tak
	Skrypt	tak
TELNET agent		
	Nazwa użytkownika	tak
	Hasło	tak
	Skrypt	tak
Zabbix trapper		
	Dozwolone hosty	tak
Monitor bazy danych		
	Nazwa użytkownika	tak
	Hasło	tak
	Zapytanie SQL	tak
JMX agent		
	Punkt końcowy JMX	tak
HTTP agent		
	URL[3]	tak
	Pola zapytania	tak
	Treść żądania	tak
	Nagłówki (nazwy i wartości)	tak
	Wymagane kody statusu	tak
	Nazwa użytkownika do uwierzytelniania HTTP	tak
	Hasło do uwierzytelniania HTTP	tak
Filtry		
	Wyrażenie regularne	tak
Nadpisania		
	Filtry: wyrażenie regularne	tak
	Operacje: interwał aktualizacji (dla prototypów pozycji)	nie
	Operacje: okres przechowywania historii (dla prototypów pozycji)	nie
	Operacje: okres przechowywania trendów (dla prototypów pozycji)	nie

Odkrywanie sieci

W regule odkrywania sieci makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Interwał aktualizacji		nie
SNMP v1, v2		
	Społeczność SNMP	tak
	OID SNMP	tak
SNMP v3		
	Nazwa kontekstu	tak
	Nazwa bezpieczeństwa	tak
	Hasło uwierzytelniania	tak
	Hasło prywatności	tak

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
	OID SNMP	tak

Proxy

W konfiguracji proxy makra użytkownika mogą być używane w następujących polach:

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

Grupy proxy

W konfiguracji grupy proxy można używać makr użytkownika w następujących polach:

Location	Multiple macros/mix with text[1]
Okres przełączenia awaryjnego	nie
Minimalna liczba proxy	nie

Szablony

W konfiguracji szablonu makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Tagi[2]		
	Nazwy tagów	tak
	Wartości tagów	tak

Wyzwalacze

W konfiguracji wyzwalacza makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Nazwa	tak
Dane operacyjne	tak
Wyrażenie (tylko w stałych i parametrach funkcji; tajne makra nie są obsługiwane)	tak

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Tag do dopasowania	tak
Nazwa wpisu menu	tak
URL wpisu menu[3]	tak
Opis	tak
Tagi[2]	
Nazwy tagów	tak
Wartości tagów	tak

Scenariusz web

W konfiguracji scenariusza web makra użytkownika mogą być używane w następujących polach:

Lokalizacja		Wiele makr/mieszanie z tekstem[1]
Nazwa		tak
Interwał aktualizacji		nie
Agent		tak
HTTP proxy		tak
Zmienne (tylko wartości)		tak
Nagłówki (nazwy i wartości)		tak
Kroki		
	Nazwa	tak
	URL[3]	tak
	Zmienne (tylko wartości)	tak
	Nagłówki (nazwy i wartości)	tak
	Limit czasu	nie
	Wymagany ciąg znaków	tak
	Wymagane kody statusu	nie
Uwierzytelnianie		
	Użytkownik	tak
	Hasło	tak
	Certyfikat SSL	tak
	Plik klucza SSL	tak
	Hasło klucza SSL	tak
Tagi[2]		
	Nazwy tagów	tak
	Wartości tagów	tak

Inne lokalizacje

Oprócz lokalizacji wymienionych tutaj, makra użytkownika mogą być używane w następujących polach:

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Globalne skrypty (URL, skrypt, SSH, Telnet, IPMI), w tym tekst potwierdzenia	yes
Webhooki	
Skrypt JavaScript	no
Nazwa parametru skryptu JavaScript	no
Wartość parametru skryptu JavaScript	yes
Pulpity	
Kolumna typu danych *Text* w widżecie pulpitu *Top hosts*	yes
Parametr *Description* w widżecie pulpitu *Item value* i *Gauge*	yes
Parametr *Text* etykiety głównej/drugorzędnej w widżecie pulpitu *Honeycomb*	yes
Parametr *URL*[3] w widżecie pulpitu *URL*	yes
Monitoring > *Problemy*, *Hosty*, *Najnowsze dane*	
Kolumna *Custom text*	yes
Zbieranie danych > *Szablony*, *Hosty*	
Kolumna *Custom text*	yes
Alarmy > *Typy mediów* > *Szablony komunikatów*	
Temat	yes
Wiadomość	yes
Alarmy > *Typy mediów* > *Skrypt*	
Parametry skryptu	yes

Lokalizacja	Wiele makr/mieszanie z tekstem[1]
Alarmy > *Typy mediów* > *Typ mediów*	
Pola *Username* i *Password* dla typu mediów *Email* (gdy *Authentication* jest ustawione na ”Username and password”; zalecane są sekretne makra)	yes
Użytkownicy > *Użytkownicy* > *Media*	
Gdy aktywne	yes
Administracja > *Ogólne* > *GUI*	
Czas pracy	no
Administracja > *Ogólne* > *Limity czasu*	
Limity czasu dla typów pozycji	no
Administracja > *Ogólne* > *Łączniki*	
URL	yes
Username	yes
Password	yes
Bearer token	yes
Timeout	no
HTTP proxy	yes
Plik certyfikatu SSL	yes
Plik klucza SSL	yes
Hasło klucza SSL	yes

Pełną listę wszystkich makr obsługiwanych w Zabbix znajdziesz w sekcji obsługiwane makra.

Przypisy

[1] Jeśli dla danego miejsca nie są obsługiwane wiele makr w polu lub makra połączone z tekstem, pojedyncze makro musi wypełniać całe pole.

[2] Makra używane w nazwach i wartościach tagów są rozwijane tylko podczas procesu generowania zdarzeń.

[3] Adresy URL zawierające sekretne makro nie będą działać, ponieważ makro w nich zostanie rozwinięte jako ”******”.

[4] Makra nie są obsługiwane w kluczu pozycja elementów Zabbix trapper.

Zgodność wersji

Obsługiwane agenty

Aby zapewnić zgodność z Zabbix 8.0, agent Zabbix nie może być starszy niż w wersji 1.6 ani nowszy niż 8.0.

Może być konieczne przejrzenie konfiguracji starszych agentów, ponieważ niektóre parametry uległy zmianie, na przykład parametry związane z rejestrowaniem w wersjach wcześniejszych niż 3.0.

Aby w pełni wykorzystać najnowsze funkcje, metryki, zwiększoną wydajność i zmniejszone zużycie pamięci, używaj najnowszego obsługiwanego agenta.

Uwagi dotyczące systemu Windows XP

- W 32-bitowym systemie Windows XP nie używaj agentów Zabbix nowszych niż 6.0.x;
- W systemach Windows XP/Server 2003 nie używaj szablonów agentów nowszych niż Zabbix 4.0.x. Nowsze szablony używają angielskich liczników wydajności, które są obsługiwane dopiero od systemu Windows Vista/Server 2008.

Obsługiwane agenty 2

Starsze agenty Zabbix 2 od wersji 4.4 wzwyż są kompatybilne z Zabbix 8.0; Zabbix agent 2 nie może być nowszy niż 8.0.

Należy pamiętać, że podczas używania Zabbix agent 2 w wersjach 4.4 i 5.0 domyślnie stosowany jest interwał 10 minut do odświeżania nieobsługiwanych pozycji.

Aby w pełni wykorzystać najnowszą funkcjonalność, metryki, lepszą wydajność oraz zmniejszone zużycie pamięci, używaj najnowszego obsługiwanego agenta 2.

Obsługiwane proxy Zabbix

Aby zapewnić pełną zgodność z Zabbix 8.0, proxy muszą mieć tę samą główną wersję co serwer. Tylko proxy Zabbix 8.0.x są w pełni zgodne z serwerem Zabbix 8.0.x.

Nieaktualne proxy są częściowo obsługiwane: nadal mogą zbierać dane i wykonywać skrypty, ale nie mogą otrzymywać aktualizacji konfiguracji, takich jak nowe pozycje.

W odniesieniu do serwera Zabbix, proxy mogą być:

- *Aktualne* (proxy i serwer mają tę samą główną wersję);
- *Nieaktualne* (wersja proxy jest starsza niż wersja serwera, ale jest częściowo obsługiwana);
- *Nieobsługiwane* (wersja proxy jest starsza niż wersja poprzedniego wydania LTS serwera *lub* wersja proxy jest nowsza niż główna wersja serwera).

Przykłady:

Wersja serwera	Wersja proxy *aktualna*	Wersja proxy *nieaktualna*	Wersja proxy *nieobsługiwana*
6.4	6.4	6.0, 6.2	Starsza niż 6.0; nowsza niż 6.4
7.0	7.0	6.0, 6.2, 6.4	Starsza niż 6.0; nowsza niż 7.0
7.2	7.2	7.0	Starsza niż 7.0; nowsza niż 7.2
7.4	7.4	7.0	Starsza niż 7.0; nowsza niż 7.4
8.0	8.0	7.0, 7.2, 7.4	Starsza niż 7.0; nowsza niż 8.0

Funkcjonalność obsługiwana przez proxy:

Wersja proxy	Aktualizacja danych	Aktualizacja konfiguracji	Zadania
Aktualna	Tak	Tak	Tak
Nieaktualna	Tak	Nie	Polecenia zdalne (np. skrypty powłoki); Natychmiastowe sprawdzanie wartości pozycji (tj. *Wykonaj teraz*); Uwaga: Wstępne przetwarzanie testów z rzeczywistą wartością nie jest obsługiwane.
Niewspierana	Nie	Nie	Nie

Ostrzeżenia dotyczące używania niekompatybilnych wersji demonów Zabbix są zapisywane w logu.

Obsługiwane pliki XML

Pliki XML nie starsze niż w wersji 1.8 są obsługiwane do importu w Zabbix 8.0.

Attention:
W formacie eksportu XML zależności wyzwalaczy są przechowywane wyłącznie według nazwy. Jeśli istnieje kilka wyzwalaczy o tej samej nazwie (na przykład o różnych poziomach ważności i wyrażeniach), pomiędzy którymi zdefiniowano zależność, ich import nie jest możliwy. Takie zależności należy ręcznie usunąć z pliku XML i dodać ponownie po imporcie.

Składnia okresu czasu

Przegląd

Aby ustawić przedział czasu, należy użyć następującego formatu:

`d-d,hh:mm-hh:mm`

gdzie symbole oznaczają:

Symbol	Opis
d	Dzień tygodnia: 1 - poniedziałek, 2 - wtorek, ... , 7 - niedziela
hh	Godziny: 00-24
mm	Minuty: 00-59

Możesz określić więcej niż jeden przedział czasu, używając separatora średnika (;):

`d-d,hh:mm-hh:mm;d-d,hh:mm-hh:mm...`

Pozostawienie przedziału czasu pustego jest równoznaczne z 1-7,00:00-24:00, co jest wartością domyślną.

Attention:
Górna granica przedziału czasu nie jest uwzględniana. Dlatego jeśli określisz 09:00-18:00, ostatnią sekundą uwzględnioną w tym przedziale czasu jest 17:59:59.

Przykłady

Godziny pracy. Od poniedziałku do piątku od 9:00 do 18:00:

`1-5,09:00-18:00`

Godziny pracy plus weekend. Od poniedziałku do piątku od 9:00 do 18:00 oraz w sobotę i niedzielę od 10:00 do 16:00:

`1-5,09:00-18:00;6-7,10:00-16:00`

Symbole jednostek

Przegląd

Praca z dużymi wartościami, takimi jak 86400, 104857600 lub 1000000, może być trudna i prowadzić do błędów. Aby uprościć konfigurację i poprawić czytelność, Zabbix obsługuje symbole jednostek (sufiksy), które działają jako mnożniki wartości.

Wyrażenia wyzwalaczy bez sufiksów:

```
last(/host/system.uptime)<86400
avg(/host/system.cpu.load,600s)<10
last(/host/vm.memory.size[available])<20971520
```

Wyrażenia wyzwalaczy z sufiksami:

```
last(/host/system.uptime)<1d
avg(/host/system.cpu.load,10m)<10
last(/host/vm.memory.size[available])<20M
```

Sufiksy mogą również uprościć konfigurację innych obiektów, takich jak pozycje, widgety itp., a także pomóc w wyświetlaniu wartości pozycji w formacie czytelnym dla człowieka.

Note:
Aby sprawdzić, czy pole konfiguracji danego obiektu obsługuje sufiksy, zawsze należy zajrzeć na odpowiednią stronę dotyczącą konfigurowanego obiektu.

Przyrostki czasu

Frontend Zabbixa obsługuje następujące przyrostki czasu w konfiguracji encji:
 s - sekundy *(w przypadku użycia działa tak samo jak wartość surowa)*;
 m - minuty;
 h - godziny;
 d - dni;
 w - tygodnie;
 M - miesiące *(tylko funkcje trendów)*;
 y - lata *(tylko funkcje trendów)*.

Przyrostki czasu są obsługiwane tylko dla liczb całkowitych. Na przykład 1h jest obsługiwane, ale 1,5h lub 1.5h nie jest obsługiwane; zamiast tego użyj 90m.

Przyrostki rozmiaru pamięci

Zabbix obsługuje następujące przyrostki rozmiaru pamięci:
 K - kilobajty
 M - megabajty
 G - gigabajty
 T - terabajty

Przyrostki wartości pozycji

Przyrostki mogą być również używane do wyświetlania numerycznych wartości pozycji w czytelnym dla człowieka formacie.

Aby to włączyć, użyj jednego z następujących przyrostków w polu *Units* podczas konfigurowania pozycji:
 B - bajty;
 Bps - bajty na sekundę;
 s - sekundy, wyświetlane z użyciem maksymalnie trzech największych niezerowych jednostek czasu;
 uptime - czas, który upłynął, w formacie hh:mm:ss lub N dni, hh:mm:ss;
 unixtime - znacznik czasu Unix, sformatowany jako yyyy.mm.dd hh:mm:ss.

Do interpretacji i wyświetlania tych przyrostków mają zastosowanie następujące dodatkowe reguły:

- Dla B i Bps Zabbix używa konwersji opartej na systemie dwójkowym (1K = 1024B), zgodnie ze standardem JEDEC.
- Dla innych jednostek (takich jak Hz, W itp.) Zabbix używa konwersji dziesiętnej (1K = 1000).
- Dla s (sekund):
  - Format obejmuje yyy mmm ddd hhh mmm sss ms; wyświetlane są tylko maksymalnie trzy największe niezerowe jednostki czasu (np. 1M 10d 4h).
  - Jeśli jednostka ma wartość zero i znajduje się między dwiema niezerowymi jednostkami, jest pomijana (np. 10d 56m zamiast 10d 0h 56m).

Gdy używane są *Units*, do wartości pozycji automatycznie stosowane są następujące przyrostki mnożnika:
 K, **M**, **G**, **T** - kilo, mega, giga, tera;
 P, **E**, **Z**, **Y** - peta, eksa, zetta, yotta *(są one stosowane tylko w frontend)*.

Aby zapobiec konwersji jednostek, użyj prefiksu ! (np. !B lub !s).

Poniższe przykłady pokazują, jak odebrane wartości pozycji są konwertowane na podstawie określonych jednostek:

```
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:
Przed Zabbix 4.0 istniała zakodowana na stałe lista wykluczeń jednostek, obejmująca `ms`, `rpm`, `RPM`, `%`. Lista ta została wycofana, dlatego poprawnym sposobem zapobiegania konwersji takich jednostek jest `!ms`, `!rpm`, `!RPM`, `!%`.

Wykonywanie poleceń

Zabbix używa wspólnej funkcjonalności dla zewnętrznych kontroli, parametrów użytkownika, pozycji system.run, niestandardowych skryptów alertów, zdalnych poleceń i globalnych skryptów.

Kroki wykonywania

Note:
Domyślnie wszystkie skrypty w Zabbix są wykonywane przy użyciu powłoki *sh* i nie ma możliwości zmiany domyślnej powłoki. Aby użyć innej powłoki, można zastosować obejście: utworzyć plik skryptu i wywołać ten skrypt podczas wykonywania polecenia.

Polecenie/skrypt jest wykonywane w podobny sposób zarówno na platformach Unix, jak i Windows:

1. Zabbix (proces nadrzędny) tworzy potok do komunikacji
2. Zabbix ustawia potok jako wyjście dla tworzonego procesu potomnego
3. Zabbix tworzy proces potomny (uruchamia polecenie/skrypt)
4. Dla procesu potomnego tworzona jest nowa grupa procesów (w Unix) lub zadanie (w Windows)
5. Zabbix odczytuje z potoku do momentu wystąpienia limitu czasu lub gdy nikt nie zapisuje do drugiego końca (WSZYSTKIE uchwyty/deskriptory plików zostały zamknięte). Należy pamiętać, że proces potomny może utworzyć więcej procesów i zakończyć działanie, zanim one zakończą działanie lub zamkną uchwyt/deskrpytor pliku.
6. Jeśli limit czasu nie został osiągnięty, Zabbix czeka, aż początkowy proces potomny zakończy działanie lub wystąpi limit czasu
7. Jeśli początkowy proces potomny zakończył działanie i limit czasu nie został osiągnięty, Zabbix sprawdza kod wyjścia początkowego procesu potomnego i porównuje go z 0 (wartość różna od zera jest uznawana za błąd wykonania, tylko dla niestandardowych skryptów alertów, zdalnych poleceń i skryptów użytkownika wykonywanych na serwer i proxy Zabbix)
8. Na tym etapie zakłada się, że wszystko zostało wykonane i całe drzewo procesów (tj. grupa procesów lub zadanie) jest kończone

Attention:
Zabbix zakłada, że polecenie/skrypt zakończyło przetwarzanie, gdy początkowy proces potomny zakończył działanie ORAZ żaden inny proces nie utrzymuje już otwartego uchwytu wyjściowego/deskriptora pliku. Gdy przetwarzanie zostanie zakończone, WSZYSTKIE utworzone procesy są kończone.

Wszystkie podwójne cudzysłowy i ukośniki odwrotne w poleceniu są poprzedzane ukośnikami odwrotnymi, a polecenie jest ujmowane w podwójne cudzysłowy.

Sprawdzanie kodu wyjścia

Kody wyjścia są sprawdzane z uwzględnieniem następujących warunków:

- Tylko dla niestandardowych skryptów alertów, zdalnych poleceń i skryptów użytkownika wykonywanych na serwerze Zabbix i proxy Zabbix.
- Każdy kod wyjścia inny niż 0 jest uznawany za niepowodzenie wykonania.
- Zawartość standardowego błędu i standardowego wyjścia dla nieudanych wykonań jest zbierana i dostępna w frontend (gdzie wyświetlany jest wynik wykonania).
- Dodatkowy wpis w logu może zostać utworzony dla zdalnych poleceń wykonywanych na agent/proxy Zabbix przez włączenie parametru LogRemoteCommands na agent/proxy.

Możliwe komunikaty frontend i wpisy w logu dla nieudanych poleceń/skryptów:

- Zawartość standardowego błędu i standardowego wyjścia dla nieudanych wykonań (jeśli występuje).
- ”Process exited with code: N.” (dla pustego wyjścia i kodu wyjścia różnego od 0).
- ”Process killed by signal: N.” (gdy proces został zakończony sygnałem, tylko w systemie Linux).
- ”Process terminated unexpectedly.” (gdy proces zakończył się z nieznanych przyczyn).

Zobacz także

- Kontrole zewnętrzne
- Parametry użytkownika
- pozycje system.run
- Niestandardowe skrypty alertów
- Polecenia zdalne
- Skrypty globalne

Przykłady ucieczki znaków

Przegląd

Ta strona zawiera przykłady poprawnego stosowania znaków ucieczki podczas używania wyrażeń regularnych w różnych kontekstach.

> **Note:**
> Podczas korzystania z konstruktora wyrażeń wyzwalacza poprawne znaki ucieczki w wyrażeniach regularnych są dodawane automatycznie.

Przykłady

Makro użytkownika z kontekstem

Wyrażenie regularne: `\.+\"[a-z]+<br>` Makro użytkownika z kontekstem: `{$MACRO:regex:"\.+\\"[a-z]+"}`

Uwaga:

- ukośniki odwrotne nie są poprzedzane znakiem ucieczki;
- cudzysłowy są poprzedzane znakiem ucieczki.

Funkcja makra wewnątrz parametru klucza pozycji

Wyrażenie regularne: `.+:(\d+)$<br>` Klucz pozycji: `net.tcp.service[tcp,,"{{$ENDPOINT}.regsub(\".+:(\d+)$\",\1)}"]`

Uwaga:

- wyrażenie regularne wewnątrz funkcji makra `regsub` jest ujęte w podwójny cudzysłów (ponieważ zawiera zamykający nawias);
- cudzysłowy wokół wyrażenia regularnego są poprzedzane znakiem ucieczki (ponieważ cały trzeci parametr pozycji jest ujęty w podwójny cudzysłów);
- trzeci parametr klucza pozycji jest ujęty w podwójny cudzysłów, ponieważ zawiera przecinek.

Funkcja makra LLD

Wyrażenie regularne: `\.+\"([a-z]+)<br>` Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}`

Uwaga:

- ukośniki odwrotne nie są poprzedzane znakiem ucieczki;
- cudzysłowy są poprzedzane znakiem ucieczki.

Funkcja makra LLD wewnątrz kontekstu makra użytkownika

Wyrażenie regularne: `\.+\"([a-z]+)<br>` Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Makro użytkownika z kontekstem: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"[a-z]+\", \1)}"}`

Uwaga:

- sposób poprzedzania znakiem ucieczki ukośników odwrotnych dla LLD nie zmienia się;
- przy wstawianiu makra LLD do kontekstu makra użytkownika należy umieścić je w ciągu znaków:

1. wokół wyrażenia makra dodawane są cudzysłowy;
2. cudzysłowy są poprzedzane znakiem ucieczki; łącznie wprowadzane są 3 nowe ukośniki odwrotne.

Parametr tekstowy funkcji (dowolnej)

Jako przykład użyto `concat`.

Zawartość ciągu: `\.+\"[a-z]+<br>` Wyrażenie: `concat("abc", "\\.\\\"[a-z]+")`

Uwaga:

- parametry tekstowe wymagają poprzedzania znakiem ucieczki zarówno ukośników odwrotnych, jak i cudzysłowów.

Funkcja makra LLD wewnątrz parametru tekstowego funkcji

Wyrażenie regularne: `\.+\"([a-z]+)<br>` Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Wyrażenie: `concat("abc, "{{#MACRO}.iregsub(\"\\.+\\\\\"([a-z]+)\", \\1)}")`

Uwaga:

- parametry tekstowe wymagają poprzedzania znakiem ucieczki zarówno ukośników odwrotnych, jak i cudzysłowów;
- dodawana jest kolejna warstwa poprzedzania znakiem ucieczki, ponieważ makro zostanie rozwiązane dopiero po usunięciu cudzysłowów z ciągu;

Makro użytkownika z kontekstem wewnątrz parametru tekstowego funkcji

Wyrażenie regularne: `\.+\"[a-z]+<br>` Makro użytkownika z kontekstem: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Wyrażenie: `concat("abc, "{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}")`

Uwaga:

- podobnie jak w poprzednim przykładzie potrzebna jest dodatkowa warstwa poprzedzania znakiem ucieczki;
- ukośniki odwrotne i cudzysłowy są poprzedzane znakiem ucieczki tylko dla najwyższego poziomu poprzedzania znakiem ucieczki (ponieważ jest to parametr tekstowy).

Funkcja makra LLD wewnątrz kontekstu makra użytkownika wewnątrz funkcji

Wyrażenie regularne: `\.+\"([a-z])+<br>` Makro LLD: `{{#MACRO}.iregsub("\.+\\"([a-z]+)", \1)}<br>` Makro użytkownika z kontekstem: `{$MACRO:"{{#MACRO}.iregsub(\"\.+\\\"([a-z]+)\", \1)}"}<br>` Wyrażenie: `concat("abc, "{$MACRO:\"{{#MACRO}.iregsub(\\\"\.+\\\\\\\"([a-z]+)\\\", \\1)}\"}")`

Zwróć uwagę na trzy warstwy poprzedzania znakiem ucieczki:

1. Dla funkcji makra LLD, bez poprzedzania znakiem ucieczki ukośników odwrotnych;
2. Dla makra użytkownika z kontekstem, bez poprzedzania znakiem ucieczki ukośników odwrotnych;
3. Dla parametru tekstowego funkcji, z poprzedzaniem znakiem ucieczki ukośników odwrotnych.

Makro użytkownika z kontekstem bezpośrednio wewnątrz ciągu

Wyrażenie regularne: `\.+\"[a-z]+<br>` Makro użytkownika z kontekstem: `{$MACRO:regex:"\.+\\"[a-z]+"}<br>` Wewnątrz ciągu w pewnym wyrażeniu, na przykład: `func(arg1, arg2, arg3)="{$MACRO:regex:\"\\.+\\\\\"[a-z]+\"}"`

Uwaga:

- ciągi również wymagają poprzedzania znakiem ucieczki ukośników odwrotnych;
- ciągi również wymagają poprzedzania znakiem ucieczki cudzysłowów;
- ponownie jest to przypadek z 2 poziomami poprzedzania znakiem ucieczki:

1. Poprzedzanie znakiem ucieczki dla kontekstu makra użytkownika bez poprzedzania znakiem ucieczki ukośników odwrotnych;
2. Poprzedzanie znakiem ucieczki wynikające z tego, że jest to ciąg, z poprzedzaniem znakiem ucieczki ukośników odwrotnych.

Czym jest Zabbix

Zabbix został stworzony przez Aleksieja Władyszewa i jest obecnie aktywnie rozwijany oraz wspierany przez firmę Zabbix SIA.

Zabbix to rozproszone rozwiązanie do monitorowania typu open source klasy korporacyjnej.

Zabbix to oprogramowanie monitorujące liczne parametry sieci oraz stan i integralność serwerów, maszyn wirtualnych, aplikacji, usług, baz danych, witryn internetowych, chmury i innych elementów. Zabbix korzysta z elastycznego mechanizmu powiadomień, który umożliwia użytkownikom konfigurowanie alertów opartych na wiadomościach e-mail dla praktycznie każdego zdarzenia. Umożliwia to szybką reakcję na problemy z serwerami. Zabbix oferuje doskonałe funkcje raportowania i wizualizacji danych na podstawie przechowywanych danych. Dzięki temu Zabbix doskonale nadaje się do planowania pojemności.

Zabbix obsługuje zarówno odpytywanie, jak i pułapkowanie. Dostęp do wszystkich raportów i statystyk Zabbix, a także parametrów konfiguracji, można uzyskać za pośrednictwem frontend. Frontend oparty na sieci WWW zapewnia możliwość sprawdzenia stanu sieci i kondycji serwerów z dowolnego miejsca. Po prawidłowej konfiguracji Zabbix może odgrywać ważną rolę w monitorowaniu infrastruktury IT. Dotyczy to zarówno małych organizacji z kilkoma serwerami, jak i dużych firm z dużą liczbą serwerów.

Zabbix jest dostępny bezpłatnie. Zabbix został napisany i jest rozpowszechniany na licencji AGPL-3.0. Oznacza to, że jego kod źródłowy jest swobodnie rozpowszechniany i dostępny publicznie.

Dostępna jest pomoc techniczna świadczona przez firmę Zabbix i jej partnerów na całym świecie.

Dowiedz się więcej o funkcjach Zabbix.

Użytkownicy Zabbixa Wiele organizacji różnej wielkości na całym świecie polega na Zabbixie jako podstawowej platformie monitorowania.

Architektura Zabbix składa się z kilku głównych komponentów oprogramowania. Ich zadania opisano poniżej.

Serwer

Zabbix server jest centralnym komponentem, do którego agenty raportują informacje i statystyki dotyczące dostępności oraz integralności. Serwer jest centralnym repozytorium, w którym przechowywane są wszystkie dane konfiguracyjne, statystyczne i operacyjne.

Przechowywanie bazy danych

Wszystkie informacje konfiguracyjne, a także dane zbierane przez Zabbix, są przechowywane w bazie danych.

Interfejs webowy

Aby zapewnić łatwy dostęp do Zabbix z dowolnego miejsca i z dowolnej platformy, udostępniany jest interfejs oparty na przeglądarce. Interfejs jest częścią serwera Zabbix i zwykle (ale niekoniecznie) działa na tym samym fizycznym komputerze, na którym działa serwer.

Proxy

Zabbix proxy może zbierać dane o wydajności i dostępności w imieniu serwera Zabbix. Proxy jest opcjonalnym elementem wdrożenia Zabbix; jednak może być bardzo korzystny do rozłożenia obciążenia pojedynczego serwera Zabbix.

Agent

Agenty Zabbix są wdrażane na monitorowanych celach, aby aktywnie monitorować lokalne zasoby i aplikacje oraz raportować zebrane dane do serwera Zabbix. Od Zabbix 4.4 dostępne są dwa typy agentów: Zabbix agent (lekki, obsługiwany na wielu platformach, napisany w C) oraz Zabbix agent 2 (bardzo elastyczny, łatwo rozszerzalny za pomocą wtyczek, napisany w Go).

Przepływ danych Ponadto ważne jest, aby zrobić krok wstecz i spojrzeć na ogólny przepływ danych w Zabbix. Aby utworzyć pozycję, która zbiera dane, należy najpierw utworzyć host. Na drugim końcu spektrum Zabbix, aby utworzyć wyzwalacz, najpierw trzeba mieć pozycję. Aby utworzyć akcję, trzeba mieć wyzwalacz. Jeśli więc chcesz otrzymać alert, że obciążenie CPU na *Server X* jest zbyt wysokie, musisz najpierw utworzyć wpis hosta dla *Server X*, następnie pozycję do monitorowania jego CPU, potem wyzwalacz, który aktywuje się, jeśli CPU jest zbyt wysokie, a na końcu akcję, która wyśle Ci wiadomość e-mail. Choć może się to wydawać dużą liczbą kroków, przy użyciu szablonów tak naprawdę tak nie jest. Jednak dzięki takiej konstrukcji możliwe jest stworzenie bardzo elastycznej konfiguracji.

Funkcje Zabbix to wysoce zintegrowane rozwiązanie do monitorowania sieci, oferujące wiele funkcji w jednym pakiecie.

Zbieranie danych

- sprawdzanie dostępności i wydajności
- obsługa SNMP (zarówno trapping, jak i polling), IPMI, JMX, monitorowanie VMware
- niestandardowe sprawdzenia
- zbieranie wymaganych danych w niestandardowych interwałach
- realizowane przez serwer/proxy oraz przez agentów

Elastyczne definicje progów

- można definiować bardzo elastyczne progi problemów, zwane wyzwalaczami, odwołujące się do wartości z bazy danych backendu

Wysoce konfigurowalne alertowanie

- wysyłanie powiadomień można dostosować do harmonogramu eskalacji, odbiorcy i typu medium
- powiadomienia można uczynić bardziej znaczącymi i pomocnymi za pomocą zmiennych makro
- automatyczne działania obejmują zdalne polecenia

Wykresy w czasie rzeczywistym

- monitorowane pozycje są natychmiast przedstawiane na wykresach dzięki wbudowanej funkcjonalności tworzenia wykresów

Możliwości monitorowania WWW

- Zabbix może śledzić ścieżkę symulowanych kliknięć myszą na stronie WWW i sprawdzać funkcjonalność oraz czas odpowiedzi

Rozbudowane opcje wizualizacji

- możliwość tworzenia niestandardowych wykresów, które mogą łączyć wiele pozycji w jednym widoku
- mapy sieci
- pokazy slajdów w widoku przypominającym pulpit nawigacyjny
- raporty
- widok wysokiego poziomu (biznesowy) monitorowanych zasobów

Przechowywanie danych historycznych

- dane przechowywane w bazie danych
- konfigurowalna historia
- wbudowana procedura housekeeping

Łatwa konfiguracja

- dodawanie monitorowanych urządzeń jako hostów
- hosty są przejmowane do monitorowania po zapisaniu w bazie danych
- stosowanie szablonów do monitorowanych urządzeń

Użycie szablonów

- grupowanie sprawdzeń w szablonach
- szablony mogą dziedziczyć po innych szablonach

Odkrywanie sieci

- automatyczne wykrywanie urządzeń sieciowych
- automatyczna rejestracja agentów
- wykrywanie systemów plików, interfejsów sieciowych i OID SNMP

Szybki frontend WWW

- frontend WWW oparty na PHP
- dostępny z dowolnego miejsca
- można poruszać się po nim klikając
- dziennik audytu

API Zabbix

- API Zabbix udostępnia programowalny interfejs do Zabbix do masowych operacji, integracji z oprogramowaniem firm trzecich i innych zastosowań.

System uprawnień

- bezpieczne uwierzytelnianie użytkowników
- dostęp niektórych użytkowników może być ograniczony do określonych widoków

W pełni funkcjonalny i łatwo rozszerzalny agent

- wdrażany na monitorowanych celach
- może być wdrażany zarówno w systemach Linux, jak i Windows

Demony binarne

- napisane w C, dla wydajności i niewielkiego zużycia pamięci
- łatwe do przenoszenia

Gotowy na złożone środowiska

- zdalne monitorowanie ułatwione dzięki użyciu proxy Zabbix

Słownik W tej sekcji można poznać znaczenie niektórych terminów często używanych w Zabbix.

host

- dowolne urządzenie fizyczne lub wirtualne, aplikacja, usługa albo inny logicznie powiązany zbiór monitorowanych parametrów.

grupa hostów

- logiczne pogrupowanie hostów. Grupy hostów są używane podczas przypisywania uprawnień dostępu do hostów różnym grupom użytkowników.

pozycja

- konkretna porcja danych, którą chcesz otrzymywać z hosta, czyli metryka danych.

wstępne przetwarzanie wartości

- *przekształcenie otrzymanej wartości metryki* przed zapisaniem jej w bazie danych.

wyzwalacz

- *wyrażenie logiczne definiujące próg problemu i używane do „oceny" danych otrzymywanych w pozycjach.*

Gdy otrzymane dane przekroczą próg, wyzwalacze przechodzą ze stanu *OK* do stanu *Problem*. Gdy otrzymane dane spadną poniżej progu, wyzwalacze pozostają w stanie *OK* lub do niego powracają.

szablon

- *zestaw encji (pozycji, wyzwalaczy, wykresów, reguł wykrywania niskiego poziomu, scenariuszy internetowych) gotowych do zastosowania na jednym lub kilku hostach.*

Zadaniem szablonów jest przyspieszenie wdrażania zadań monitorowania na hoście, a także ułatwienie stosowania zbiorczych zmian w zadaniach monitorowania. Szablony są bezpośrednio powiązane z poszczególnymi hostami.

grupa szablonów

- *logiczne pogrupowanie szablonów. Grupy szablonów są używane podczas przypisywania uprawnień dostępu do szablonów różnym grupom użytkowników.*

zdarzenie

- *pojedyncze wystąpienie czegoś, co wymaga uwagi, na przykład zmiana stanu wyzwalacza albo przeprowadzenie wykrywania lub automatycznej rejestracji agenta.*

znacznik zdarzenia

- *wstępnie zdefiniowany znacznik zdarzenia.* Może być używany w korelacji zdarzeń, szczegółowym przydzielaniu uprawnień itp.

korelacja zdarzeń

- *metoda elastycznego i precyzyjnego korelowania problemów z ich rozwiązaniem.*

Można na przykład zdefiniować, że problem zgłoszony przez jeden wyzwalacz może zostać rozwiązany przez inny wyzwalacz, który może nawet korzystać z innej metody zbierania danych.

problem

- *wyzwalacz znajdujący się w stanie „Problem".*

aktualizacja problemu

- *opcje zarządzania problemami udostępniane przez Zabbix, takie jak dodanie komentarza, potwierdzenie, zmiana ważności lub ręczne zamknięcie.*

akcja

- *wstępnie zdefiniowany sposób reagowania na zdarzenie.*

Akcja składa się z operacji (np. wysłania powiadomienia) oraz warunków (*kiedy* operacja jest wykonywana).

eskalacja

- *niestandardowy scenariusz wykonywania operacji w ramach akcji; sekwencja wysyłania powiadomień lub wykonywania zdalnych poleceń.*

makro

- *symbol zastępczy zmiennej, który jest rozwijany do określonej wartości zależnie od kontekstu.*

medium

- *sposób dostarczania powiadomień; kanał dostarczania.*

powiadomienie

- *wiadomość o określonym zdarzeniu wysłana do użytkownika za pośrednictwem wybranego kanału medium.*

zdalne polecenie

- *wstępnie zdefiniowane polecenie, które jest automatycznie wykonywane na monitorowanym hoście po wystąpieniu określonego warunku.*

scenariusz internetowy

- *jedno lub kilka żądań HTTP służących do sprawdzania dostępności witryny internetowej.*

frontend

- *interfejs internetowy dostarczany wraz z Zabbix.*

pulpit nawigacyjny

- *konfigurowalna sekcja interfejsu internetowego wyświetlająca podsumowania i wizualizacje* ważnych informacji w jednostkach wizualnych nazywanych widżetami.

widżet

- *jednostka wizualna wyświetlająca informacje określonego rodzaju i pochodzące z określonego źródła* (podsumowanie, mapa, wykres, zegar itp.), używana na pulpicie nawigacyjnym.

Zabbix API

- *Zabbix API umożliwia korzystanie z protokołu JSON RPC w celu tworzenia, aktualizowania i pobierania obiektów Zabbix (takich jak hosty, pozycje, wykresy i inne) oraz wykonywania dowolnych innych zadań niestandardowych.*

serwer Zabbix

- *centralny proces oprogramowania Zabbix, który wykonuje monitorowanie, współdziała z proxy Zabbix i agentami, oblicza wyzwalacze, wysyła powiadomienia oraz stanowi centralne repozytorium danych.*

proxy Zabbix

- *proces, który może zbierać dane w imieniu serwera Zabbix, odciążając serwer z części zadań przetwarzania.*

agent Zabbix

- *proces wdrażany na monitorowanych obiektach w celu aktywnego monitorowania lokalnych zasobów i aplikacji.*

agent Zabbix 2

- *agent Zabbix nowej generacji służący do aktywnego monitorowania lokalnych zasobów i aplikacji, umożliwiający korzystanie z niestandardowych wtyczek do monitorowania.*

> **Attention:**
> Ponieważ agent Zabbix 2 ma wiele funkcji wspólnych z agentem Zabbix, termin „agent Zabbix" w dokumentacji oznacza zarówno agenta Zabbix, jak i agenta Zabbix 2, jeśli ich działanie jest takie samo. Agent Zabbix 2 jest wymieniany z nazwy tylko wtedy, gdy jego funkcjonalność się różni.

szyfrowanie

- *obsługa szyfrowanej komunikacji między komponentami Zabbix (serwerem, proxy, agentem, narzędziami zabbix_sender i zabbix_get)* z użyciem protokołu Transport Layer Security (TLS).

automatyczna rejestracja agenta

- *automatyczny proces, w ramach którego agent Zabbix jest rejestrowany* jako host i rozpoczyna monitorowanie.

wykrywanie sieci

- *automatyczne wykrywanie urządzeń sieciowych*.

wykrywanie niskiego poziomu

- *automatyczne wykrywanie encji niskiego poziomu na określonym urządzeniu* (np. systemów plików, interfejsów sieciowych itp.).

reguła wykrywania niskiego poziomu

- *zestaw definicji służących do automatycznego wykrywania encji niskiego poziomu* na urządzeniu.

prototyp pozycji

- *metryka z określonymi parametrami występującymi jako zmienne, gotowa do wykrywania niskiego poziomu.* Po wykrywaniu niskiego poziomu zmienne są automatycznie zastępowane rzeczywistymi wykrytymi parametrami, a metryka automatycznie rozpoczyna zbieranie danych.

prototyp wyzwalacza

- *wyzwalacz z określonymi parametrami występującymi jako zmienne, gotowy do wykrywania niskiego poziomu.* Po wykrywaniu niskiego poziomu zmienne są automatycznie zastępowane rzeczywistymi wykrytymi parametrami, a wyzwalacz automatycznie rozpoczyna ocenianie danych.

Prototypy niektórych innych encji Zabbix są również używane w wykrywaniu niskiego poziomu: prototypy wykresów, prototypy hostów i prototypy grup hostów.

Zabbix Cloud

Zabbix Cloud

Poznaj dostępne opcje konfiguracji i mechanizmy kontroli dostępu dla organizacji oraz węzłów w Zabbix Cloud.

Należy pamiętać, że węzły Zabbix Cloud obecnie działają w oparciu o **Zabbix 7.0**.
 Szczegółowe informacje na temat Zabbix 7.0 można znaleźć w oficjalnej dokumentacji.

Konfiguracja i zarządzanie Zabbix Cloud

Rozpocznij korzystanie z Zabbix Cloud Instrukcje krok po kroku, które pomogą szybko uruchomić instancję Zabbix Cloud, w tym przeprowadzić wstępną konfigurację i podłączyć monitorowane urządzenia.

Konfiguracja węzła Dowiedz się, jak skonfigurować węzeł Zabbix Cloud, koncentrując się na dostępie, obsłudze danych i ustawieniach zasobów.

Dodawanie użytkowników Dowiedz się, jak dodawać użytkowników i przypisywać im role w organizacji Zabbix Cloud oraz jej węzłach.

Dowiedz się więcej o Zabbix Cloud

Zabbix Cloud a on-premises Porównaj różnice w zarządzaniu, skalowalności i utrzymaniu między tymi dwoma wdrożeniami i dowiedz się, które z nich może najpłynniej wspierać Twój przepływ pracy.

Poznaj Zabbix Cloud Zapoznaj się z przeglądem Zabbix Cloud, w tym z jego kluczowymi funkcjami, korzyściami oraz tym, co obejmuje subskrypcja. Odpowiedzi na często zadawane pytania dotyczące konfiguracji i użytkowania znajdziesz w FAQ.

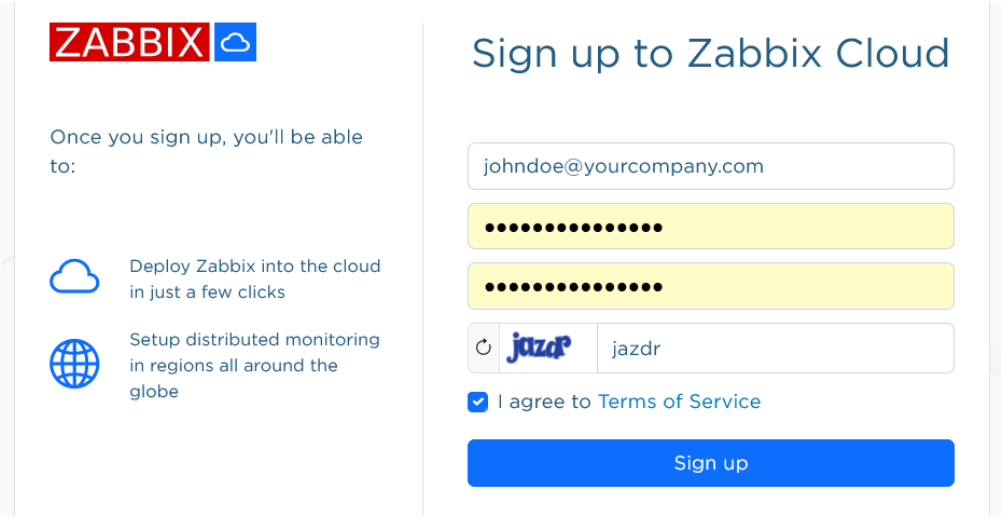
Wdróż Zabbix w chmurze

Aby uruchomić Zabbix w chmurze, wykonaj następujące czynności:

1. Zaloguj się lub zarejestruj na stronie cloud.zabbix.com.
2. Utwórz węzeł chmurowy.
3. Zacznij korzystać z Zabbix w chmurze.

Logowanie lub rejestracja Możesz zalogować się na stronie cloud.zabbix.com za pomocą istniejącego konta Google, GitHub lub Microsoft.

Możesz także przejść na stronę rejestracji, aby się zarejestrować.



Po rejestracji otrzymasz wiadomość e-mail z potwierdzeniem utworzenia nowego konta.

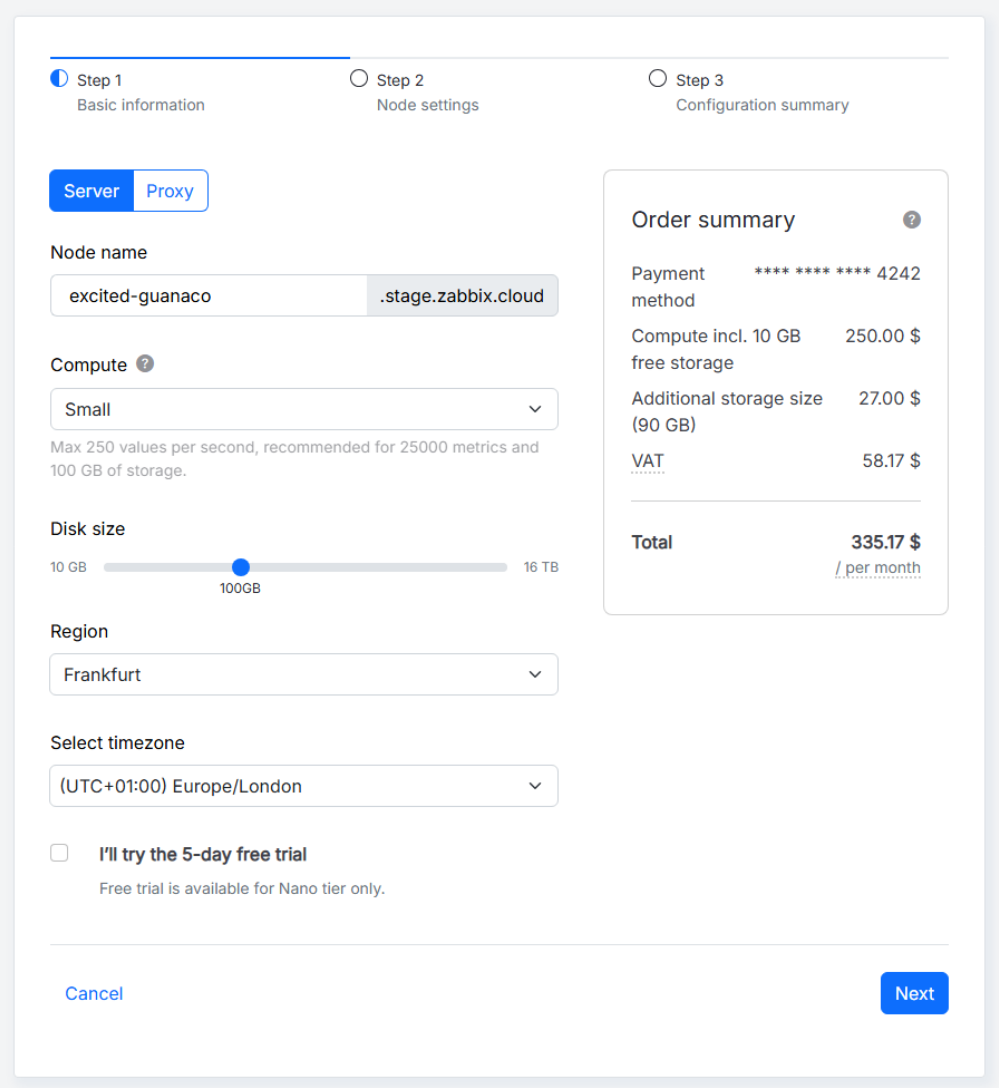
Przejdź na stronę cloud.zabbix.com, aby się zalogować. Podczas logowania zostanie wyświetlona prośba o podanie jednorazowego hasła. Hasło to zostanie wysłane w wiadomości e-mail.

Tworzenie węzła chmurowego Węzeł chmurowy to instancja Zabbix wdrożona w Zabbix Cloud. Może zostać wdrożona jako węzeł serwera lub węzeł proxy.

Po zalogowaniu zobaczysz, że dla Twojego konta domyślnie istnieje już organizacja *Private* organizacja. Nazwę domyślnej organizacji można zmienić później.

Kliknij *Create New Node*. Zostanie otwarty trzyetapowy formularz konfiguracji węzła.

W pierwszym kroku wybierzesz podstawowe informacje.



Parameter	Description
Server/Proxy	Wybierz, czy wdrożyć instancję serwera Zabbix, czy instancję proxy Zabbix.
Node name	Nazwa węzła tworzy unikatową subdomenę w formacie `<your-name>.zabbix.cloud`, która służy do uzyskiwania dostępu do usług węzła.
Compute	Wybierz moc obliczeniową dla węzła. Aby określić wymagany poziom zasobów, zobacz szczegóły cenowe. Jeśli potrzebujesz większej mocy, zawsze możesz później zwiększyć ten poziom. Przed utworzeniem węzła nieobjętego wersją próbną wymagane będzie skonfigurowanie metody rozliczeń. Dodaj informacje rozliczeniowe za pomocą menu *Billing*. Dostępna jest **bezpłatna wersja próbna** trwająca 5 dni, umożliwiająca wypróbowanie węzła warstwy *Nano*. W okresie próbnym nie zostaną naliczone opłaty, ale po jego zakończeniu węzeł zostanie usunięty, jeśli nie zostanie przekształcony w płatny węzeł.
Disk size	Suwak jest domyślnie ustawiony na zalecane minimum zależne od warstwy. Zwiększ tę wartość, jeśli przechowujesz duże ilości danych historycznych lub trendów. Pamiętaj, że większe dyski wiążą się z wyższymi miesięcznymi kosztami oraz że nie możesz samodzielnie zmniejszyć rozmiaru dysku — w celu jego zmniejszenia należy skontaktować się z pomocą techniczną.
Region	Wybierz region, w którym zostanie wdrożony węzeł.

Parameter	Description
Select timezone	Wybierz strefę czasową serwera Zabbix. Ta strefa czasowa jest używana przez serwer Zabbix i jest niezależna od stref czasowych konserwacji oraz profilu użytkownika.

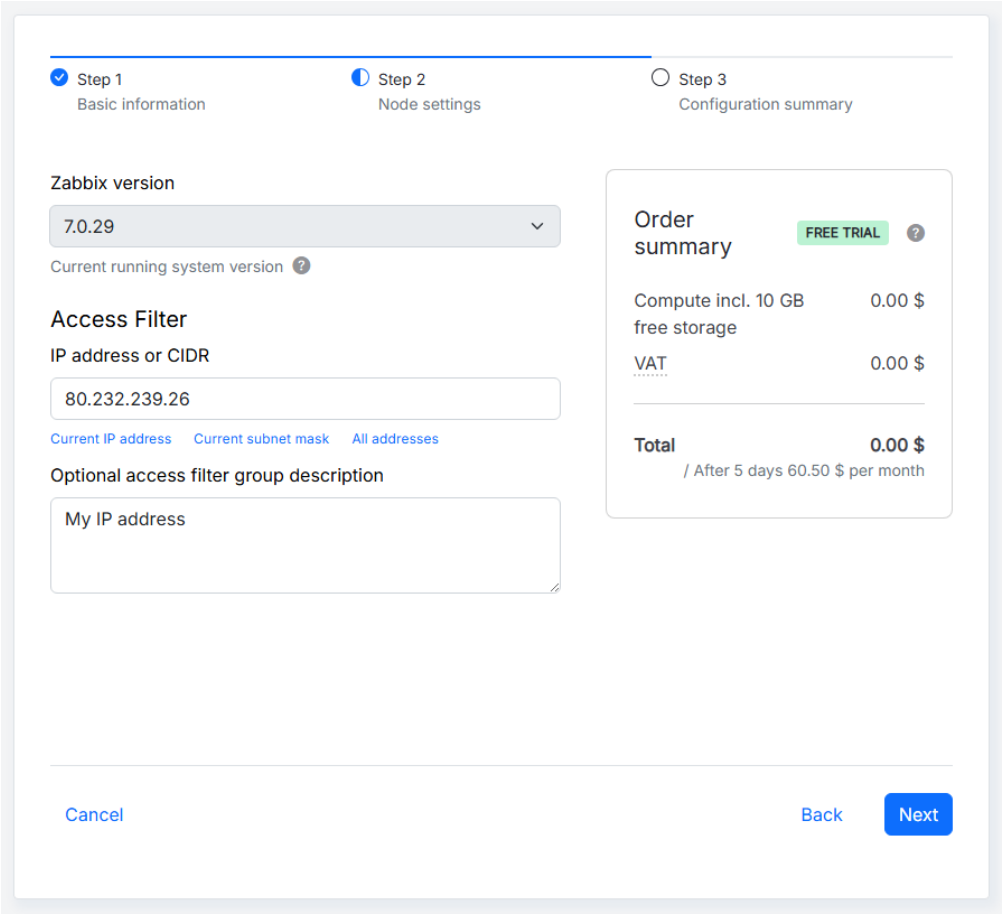
Po zakończeniu konfiguracji kliknij *Next*.

W drugim kroku wybierzesz ustawienia węzła.

Opcje konfiguracji w drugim kroku zależą od tego, czy tworzysz węzeł serwera, czy proxy.

Konfigurowanie węzła serwera

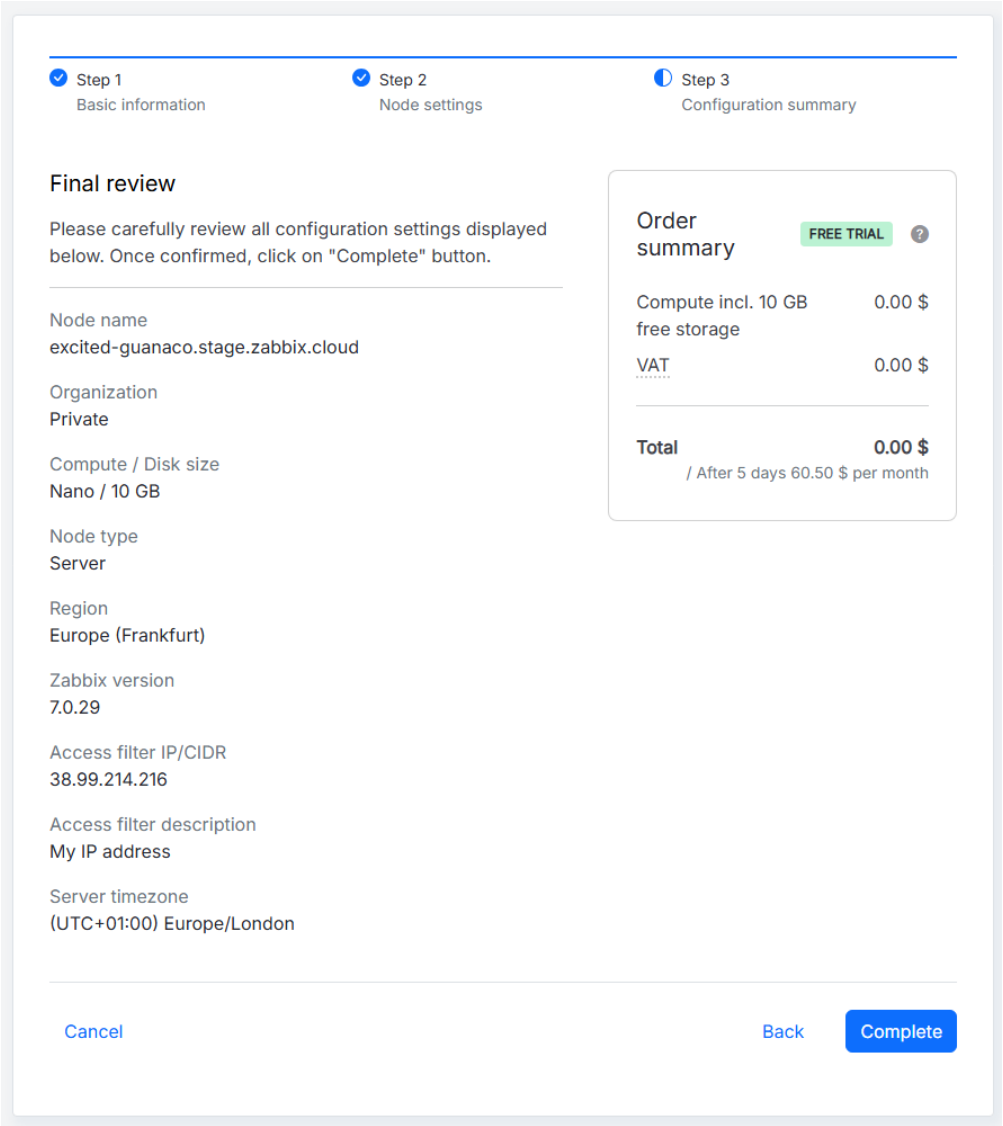
Podczas tworzenia węzła serwera dostępne są następujące parametry:



Parameter	Description
Zabbix version	Wybierz wersję Zabbix. Zabbix Cloud obsługuje najnowszą wersję LTS i automatycznie stosuje aktualizacje wersji minor.
IP address or CIDR	Określ początkowy filtr dostępu dla węzła. Wprowadź adres IP lub CIDR ręcznie albo automatycznie uzupełnij pole, wybierając odpowiednią opcję. W przypadku węzłów serwera filtr ten dotyczy połączeń z serwerem, API i frontend. Filtr można później zmienić na karcie Access filters.
Optional access filter group description	Dodaj opcjonalny opis grupy filtrów dostępu, która może zawierać wiele adresów IP lub zakresów CIDR.

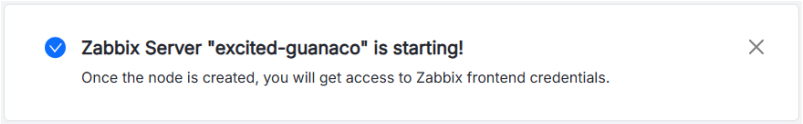
Warning:
Podczas tworzenia węzła pole *IP address or CIDR* jest wstępnie wypełniane adresem IP używanym do uzyskania dostępu do Zabbix Cloud. Możesz zmienić tę wartość, aby określić inny początkowy filtr dostępu. Po utworzeniu węzła możesz dodawać, edytować lub usuwać filtry dostępu na karcie *Access filters*.

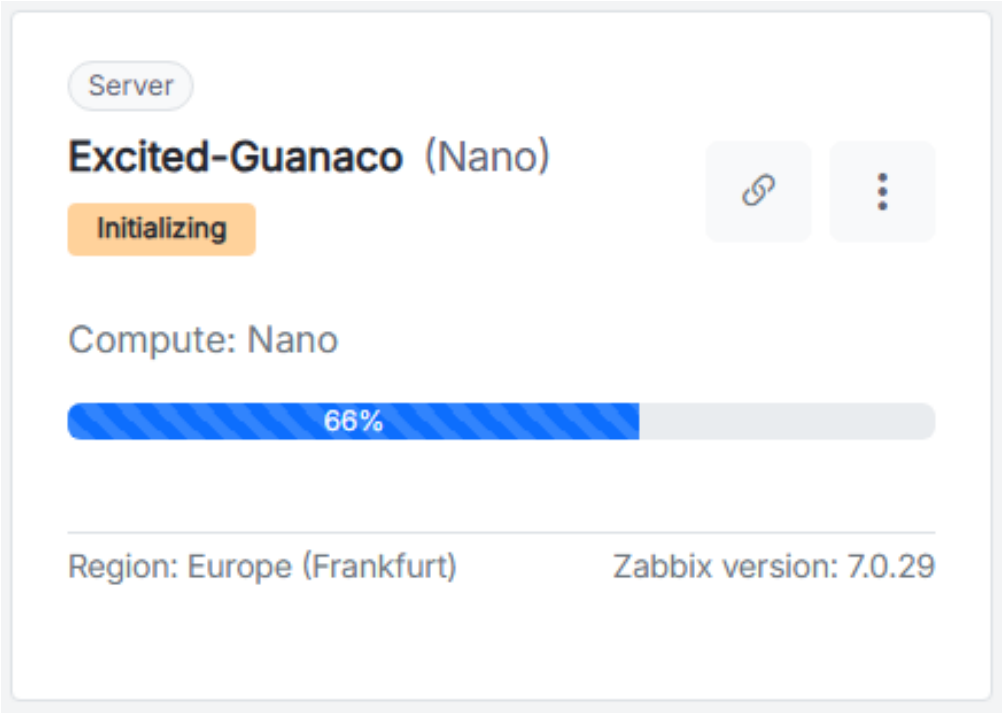
W trzecim kroku zostanie wyświetlone podsumowanie konfiguracji węzła.



Po przejrzeniu ustawień konfiguracji i potwierdzeniu, że wszystko jest poprawne, kliknij *Complete* w przypadku węzła próbnego lub *Proceed to payment* w przypadku węzła płatnego. W przeciwnym razie wróć do poprzedniego kroku, klikając przycisk *Back*, albo kliknij *Cancel*, aby rozpocząć od początku.

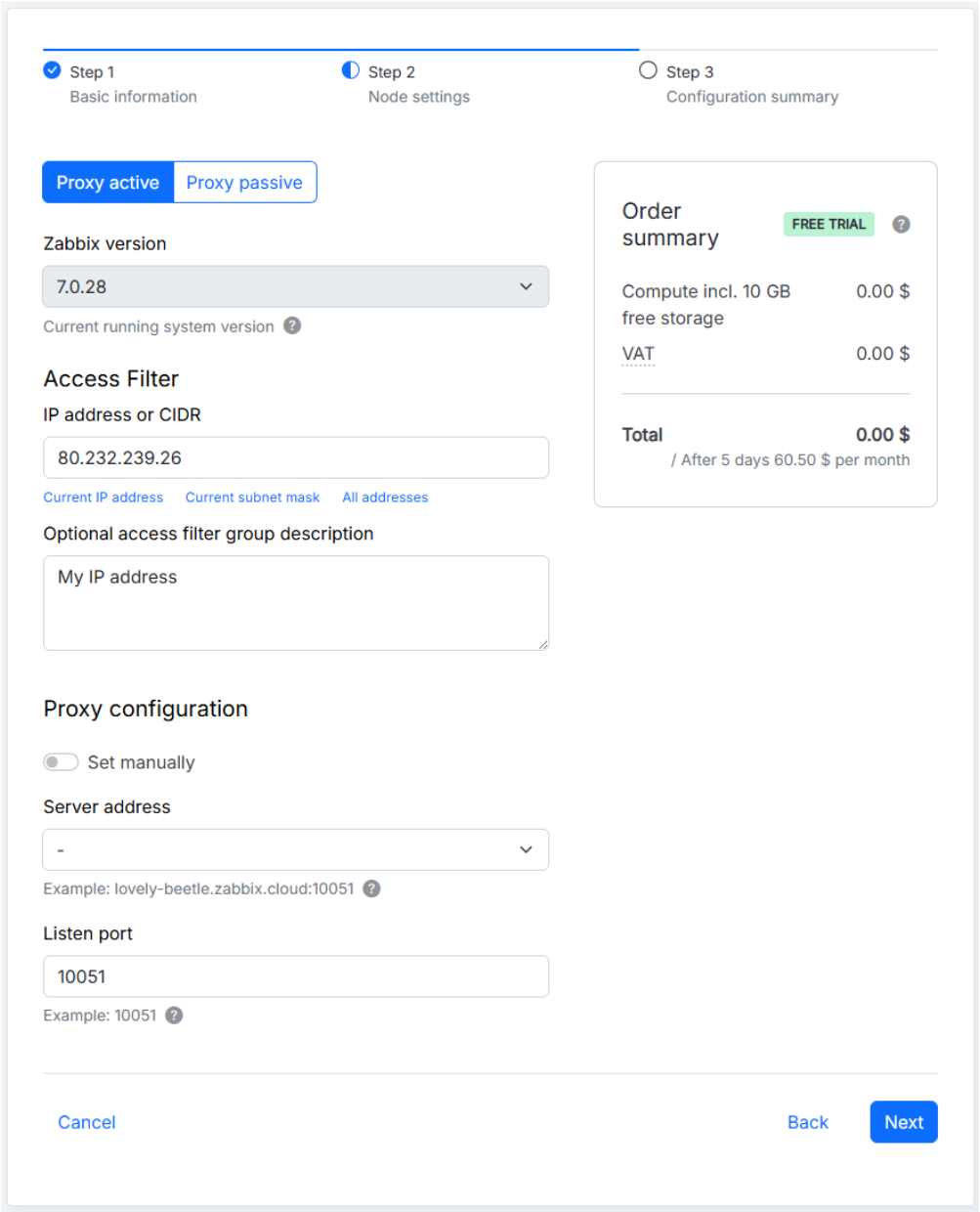
Podczas inicjalizacji węzła wyświetlany jest odpowiedni komunikat, a węzeł pojawia się już w sekcji *Organizations*.





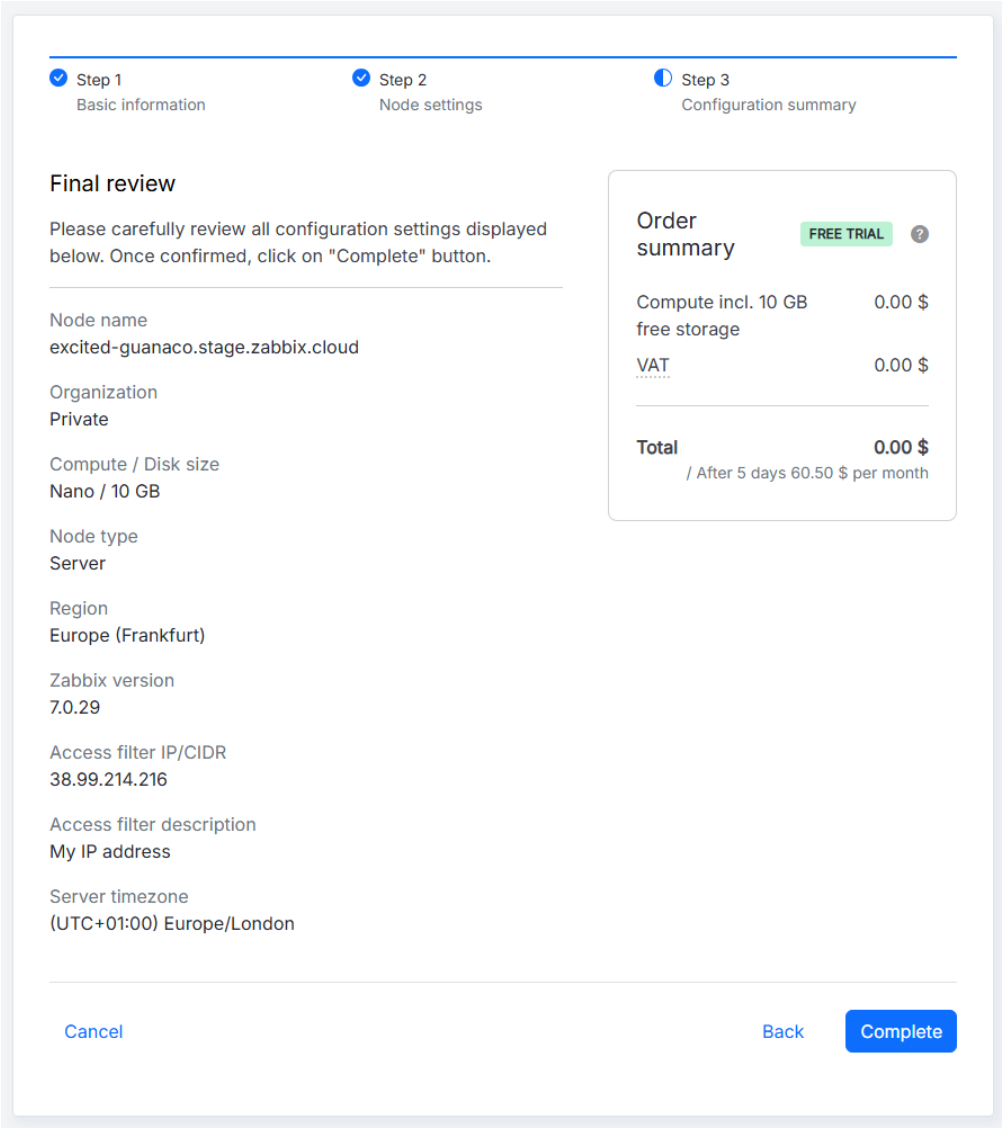
Konfigurowanie węzła proxy

Podczas tworzenia węzła proxy dostępne są następujące parametry:



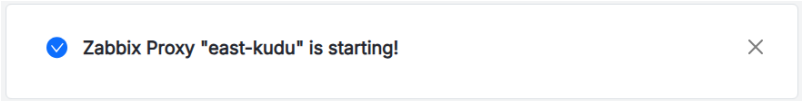
Parameter	Description
Proxy active/Proxy passive	Wybierz tryb proxy. W trybie aktywnym proxy inicjuje połączenia z serwerem Zabbix; w trybie pasywnym serwer Zabbix inicjuje połączenia z proxy.
Zabbix version	Wybierz wersję Zabbix. Zabbix Cloud obsługuje najnowszą wersję LTS i automatycznie stosuje aktualizacje wersji minor.
IP address or CIDR	Określ początkowy przychodzący filtr dostępu IP dla proxy. Wprowadź adres IP lub CIDR ręcznie albo automatycznie uzupełnij pole, wybierając odpowiednią opcję. Filtr określa, które żądania sieciowe mogą dotrzeć do proxy, i można go później zmienić na karcie Access filters.
Optional access filter group description	Dodaj opcjonalny opis grupy filtrów dostępu, która może zawierać wiele adresów IP lub zakresów CIDR.
Set manually	Wybierz sposób określenia serwera Zabbix. Gdy opcja jest wyłączona, wybierz węzeł serwera z listy dostępnych węzłów serwera. Gdy opcja jest włączona, określ adresy serwera ręcznie.
Server address	Określ adres serwera Zabbix. Gdy opcja *Set manually* jest wyłączona, wybierz węzeł serwera z listy. Gdy jest włączona, określ maksymalnie 10 adresów serwera. W przypadku aktywnego proxy określ adres IP lub nazwę DNS serwera Zabbix. W przypadku pasywnego proxy określ adres IP, nazwę DNS lub CIDR serwera Zabbix. Filtry dostępu są konfigurowane oddzielnie i nie są automatycznie aktualizowane po określeniu adresu serwera.
Listen port	Określ port, na którym proxy nasłuchuje połączeń przychodzących. Domyślny port to 10051. Port nasłuchiwania nie może zostać zmieniony po utworzeniu węzła.

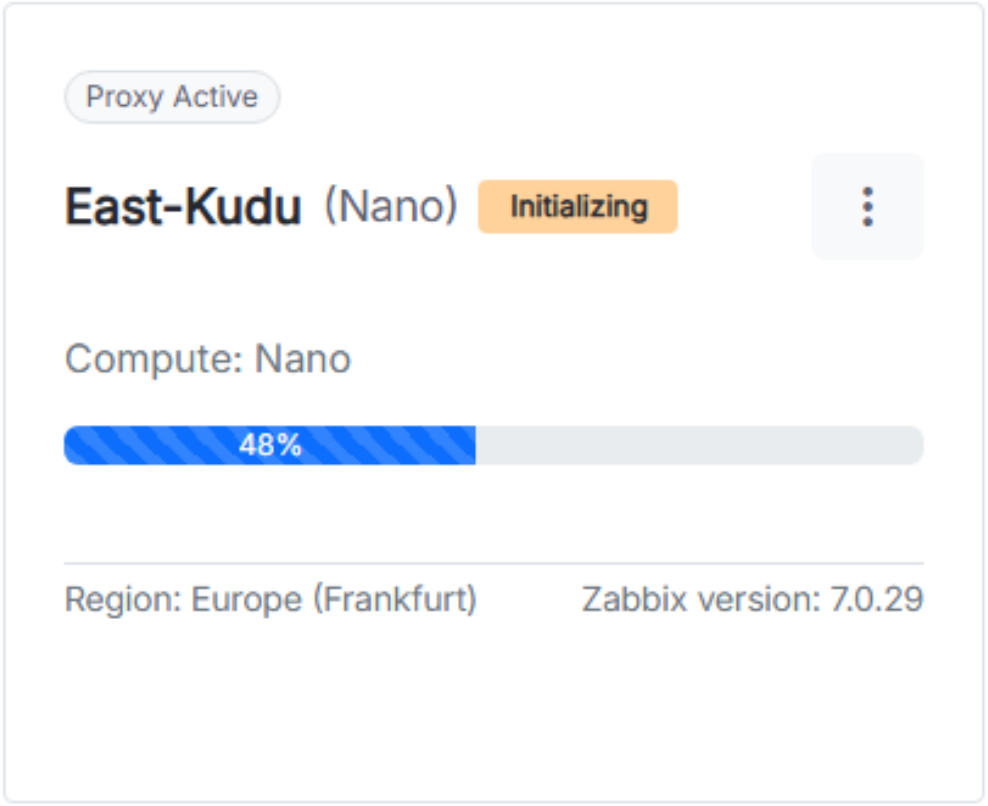
W trzecim kroku zobaczysz podsumowanie konfiguracji węzła.



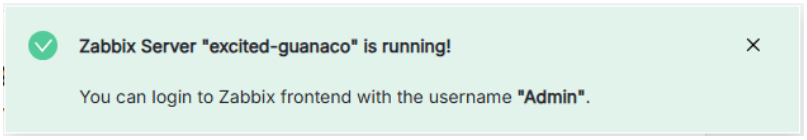
Po przejrzeniu ustawień konfiguracji i potwierdzeniu, że wszystko jest poprawne, kliknij *Complete* w przypadku węzła próbnego lub *Proceed to payment* w przypadku węzła płatnego. W przeciwnym razie wróć do poprzednich kroków, klikając przycisk *Back*, albo kliknij *Cancel*, aby rozpocząć od początku.

Podczas inicjalizacji węzła wyświetlany jest odpowiedni komunikat, a węzeł pojawia się już w sekcji *Organizations*.





Aby skonfigurować proxy Zabbix Cloud, zobacz Konfigurowanie proxy Zabbix Cloud.

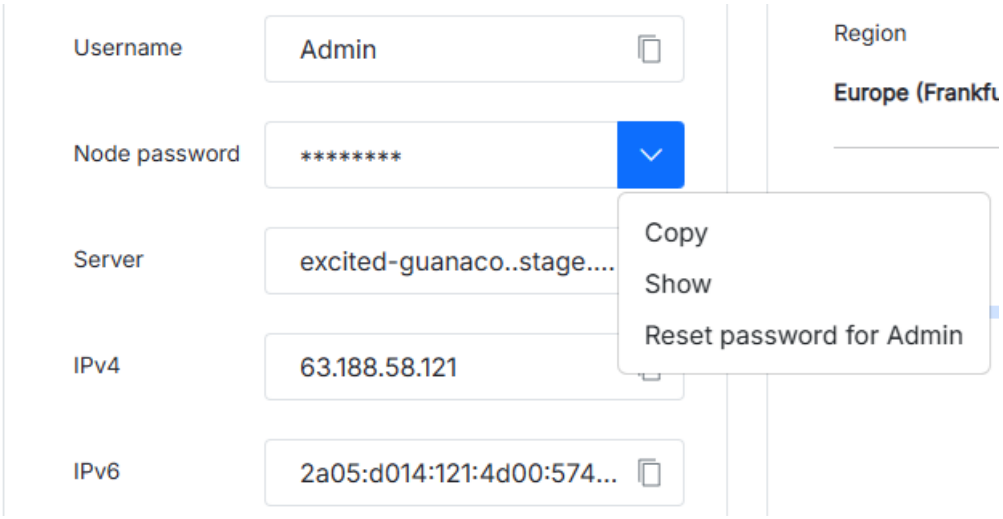


Rozpocznij korzystanie z Zabbix

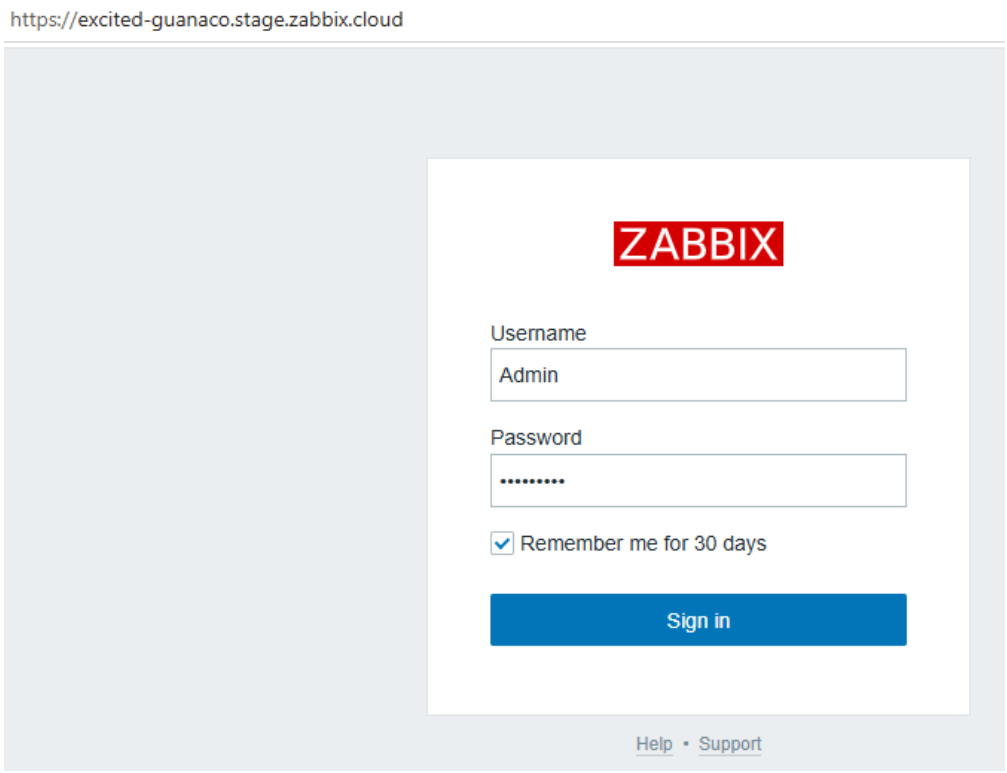
Gdy status węzła serwera zmieni się z *Initializing* na *Running*, możesz zalogować się do frontendu Zabbix.

Aby uzyskać początkowe hasło frontendu, kliknij utworzony węzeł.

Zostaniesz przeniesiony do ustawień węzła (przegląd, filtry dostępu, szyfrowanie, kopie zapasowe itd.). W sekcji **Connection info** znajdź pole *Node password* i wybierz z listy rozwijanej opcję *Reset password for Admin*. Po zresetowaniu hasła kliknij tę samą listę rozwijaną i wybierz opcję *Copy*.



Kliknij adres URL frontendu, aby otworzyć frontend Zabbix (`excited-guanaco.zabbix.cloud`).



Zaloguj się, używając następujących danych uwierzytelniających:

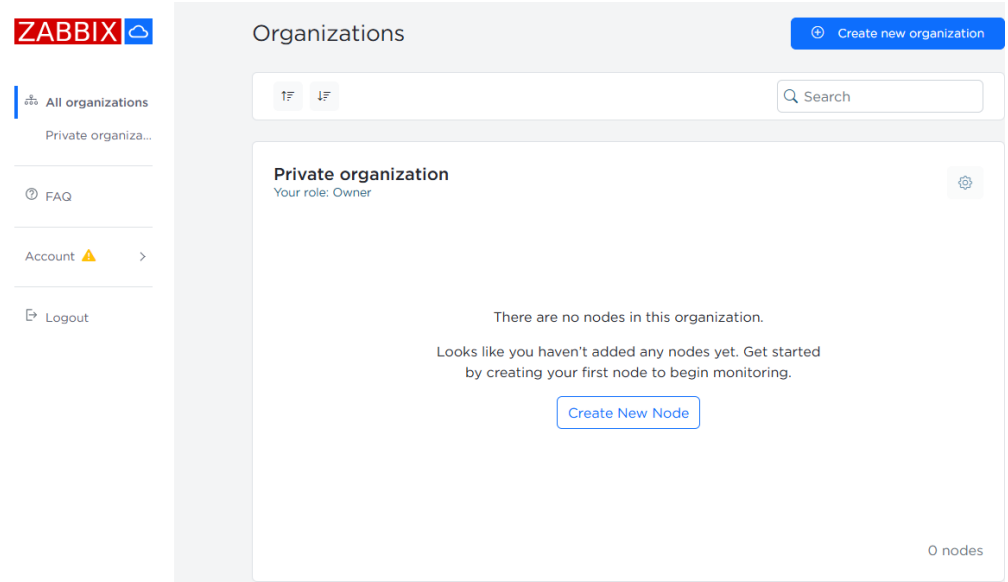
- *Username* - `Admin`
- *Password* - wprowadź skopiowane hasło

Aby rozpocząć monitorowanie, musisz zdefiniować filtry dostępu - adres IP lub adresy IP, z których można łączyć się z serwerem Zabbix Cloud. Należy zdefiniować co najmniej jeden.

Zarządzanie organizacją Organizacja jest kontenerem dla co najmniej jednego węzła. Organizacje pomagają również porządkować zasoby i delegować obowiązki, takie jak:

- Zarządzanie węzłami
- Rozliczenia

Po zalogowaniu się do Zabbix Cloud wyświetlana jest strona *Wszystkie organizacje*. Domyślnie dla Twojego konta tworzona jest organizacja prywatna.

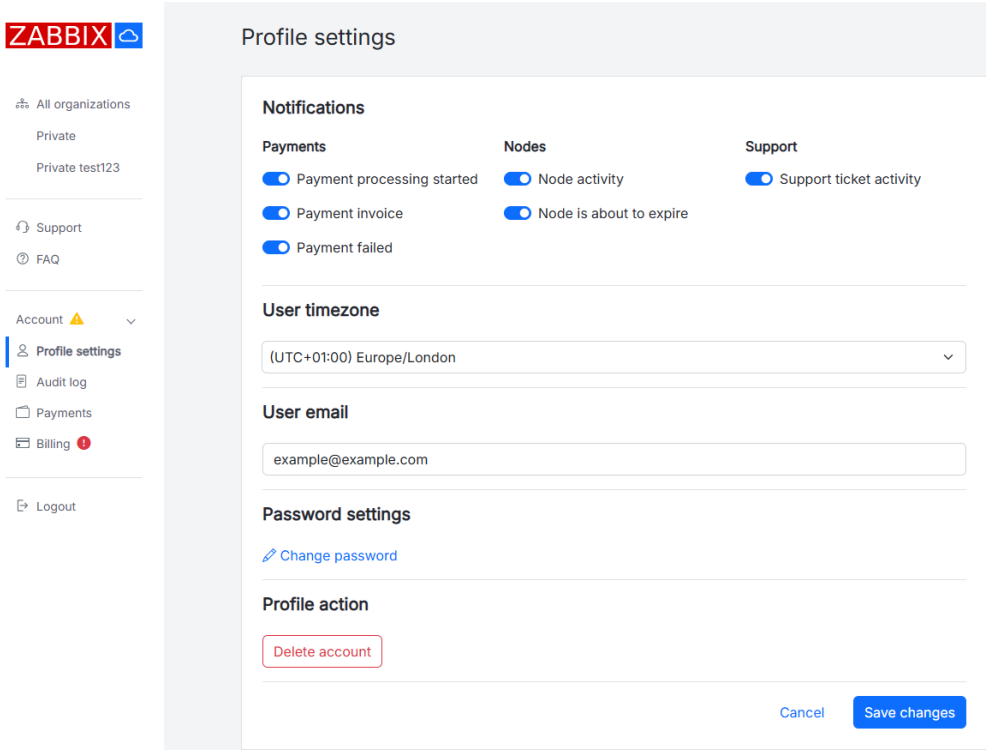


Jako właściciel tej organizacji możesz zarządzać użytkownikami, administratorami, węzłami i nazwą organizacji. Kliknij przycisk ustawień , aby uzyskać dostęp do opcji zarządzania.

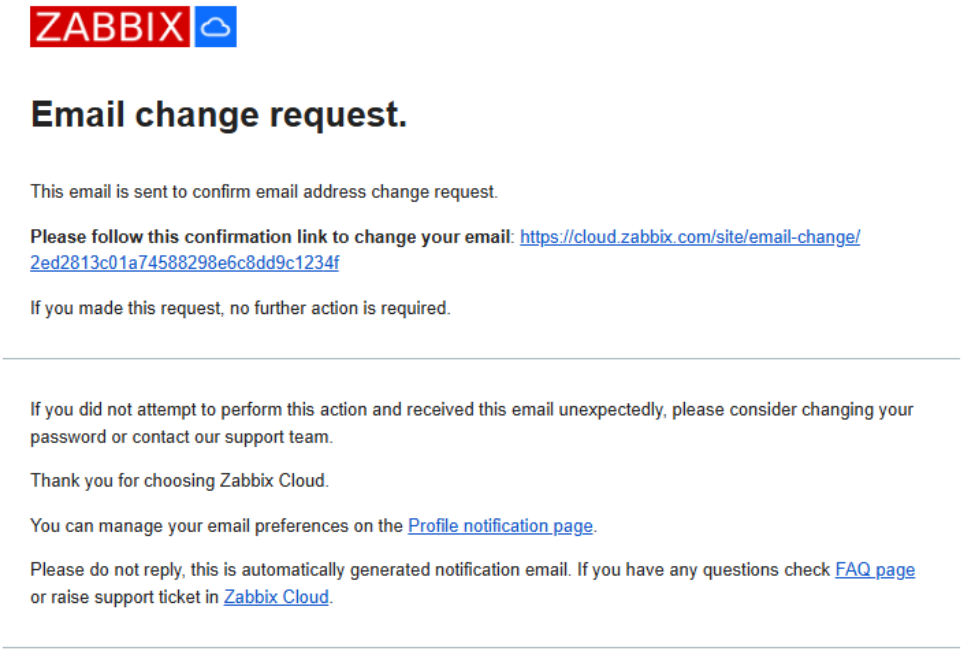
Zarządzanie kontem Zmiana adresu e-mail konta

Aby zmienić adres e-mail konta, przejdź do sekcji *Konto* > *Ustawienia profilu*.

W polu *E-mail użytkownika* wprowadź nowy adres e-mail i kliknij *Zapisz zmiany*.



Na nowy adres e-mail zostanie wysłana wiadomość z potwierdzeniem. Kliknij link zawarty w wiadomości, aby potwierdzić zmianę adresu e-mail.



Link potwierdzający otwiera stronę *Ustawienia profilu* z komunikatem potwierdzającym oraz zaktualizowanym adresem e-mail w polu E-mail użytkownika.

Po potwierdzeniu zmiany wiadomość e-mail potwierdzająca zmianę zostanie wysłana zarówno na nowy, jak i poprzedni adres e-mail. Wiadomość wysłana na poprzedni adres zawiera poprzedni i nowy adres e-mail.

Konfiguracja węzła

Przegląd

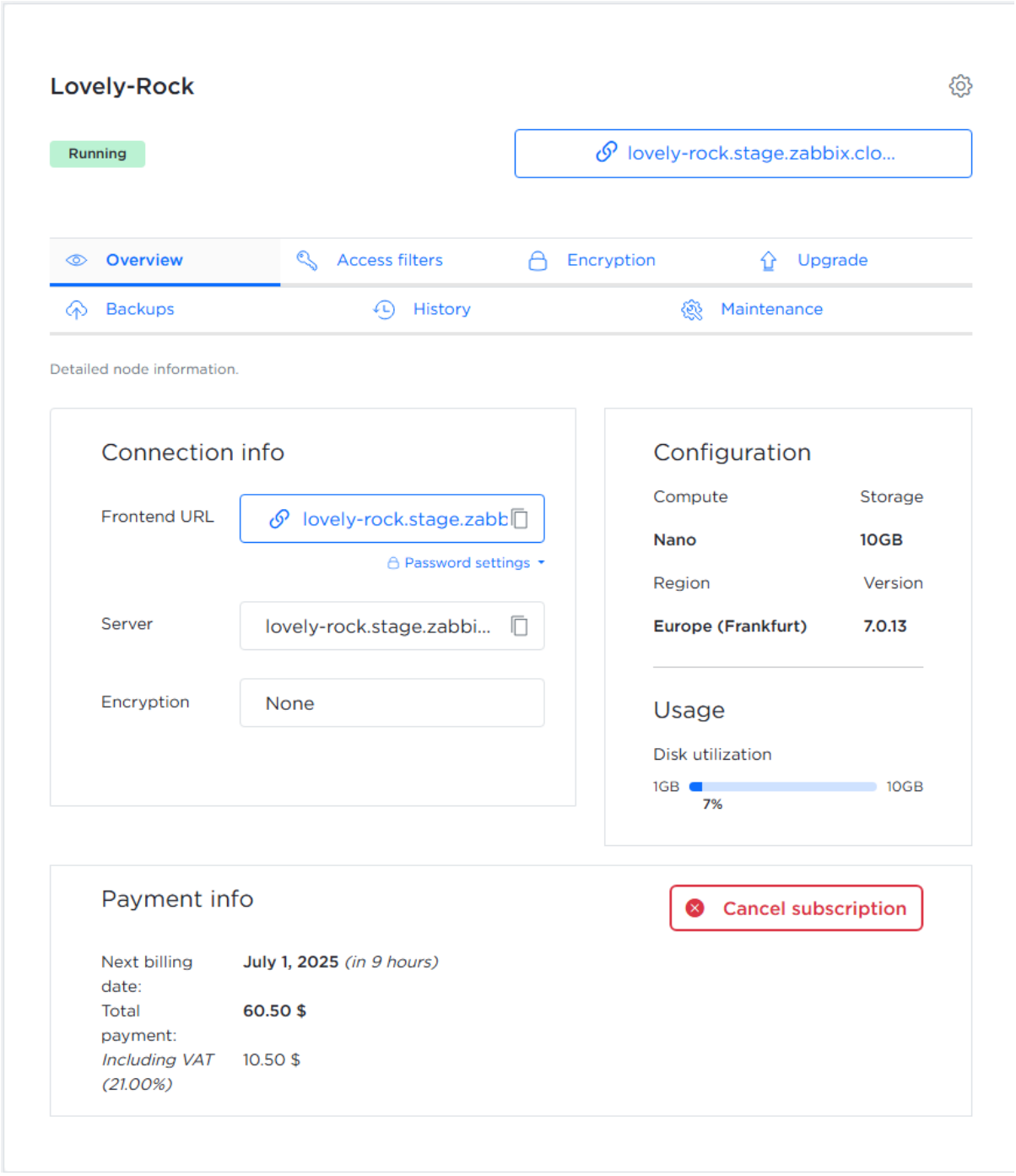
Węzeł chmurowy to komponent Zabbix wdrożony w chmurze. Węzły serwera obejmują serwer Zabbix, bazę danych i frontend, natomiast węzły proxy obejmują proxy Zabbix i bazę danych. Węzeł chmurowy ma dedykowane zasoby obliczeniowe (CPU, pamięć) oraz przestrzeń dyskową.

Użytkownicy mogą utworzyć jeden lub więcej węzłów chmurowych. Węzły chmurowe można grupować w **organizacje** (strukturalne, regionalne itp.), tworząc rozproszoną konfigurację monitorowania.

Węzły chmurowe i organizacje są tworzone za pomocą interfejsu użytkownika Zabbix Cloud. Aby uzyskać dostęp do strony konfiguracji węzła, kliknij węzeł na stronie *Wszystkie organizacje*, w swoim menu organizacji lub kliknij link *Strona konfiguracji węzła* wyświetlany po uruchomieniu węzła.

W widoku konfiguracji węzła każda karta umożliwia zarządzanie określonym aspektem węzła:

- Przegląd - wyświetlanie szczegółów węzła, otwieranie jego frontendu i zarządzanie użytkownikami węzła.
- Filtry dostępu - kontrolowanie dostępu według adresu IP lub podsieci.
- Szyfrowanie - konfigurowanie TLS przez przesyłanie certyfikatów, kluczy i opcjonalnych CRL.
- Kopie zapasowe - tworzenie, przywracanie i usuwanie migawek węzła; wyświetlanie terminu następnej zaplanowanej kopii zapasowej.
- Historia - ustawianie okresów przechowywania dzienników audytu, danych historycznych i trendów, z użyciem wstępnie zdefiniowanych jednostek oraz limitem aktualizacji 3 razy dziennie.
- Konserwacja - definiowanie cotygodniowych okien konserwacji według strefy czasowej, dnia i godziny.
- Aktualizacja - skalowanie zasobów obliczeniowych i przestrzeni dyskowej lub zarządzanie statusem wersji próbnej/płatności.

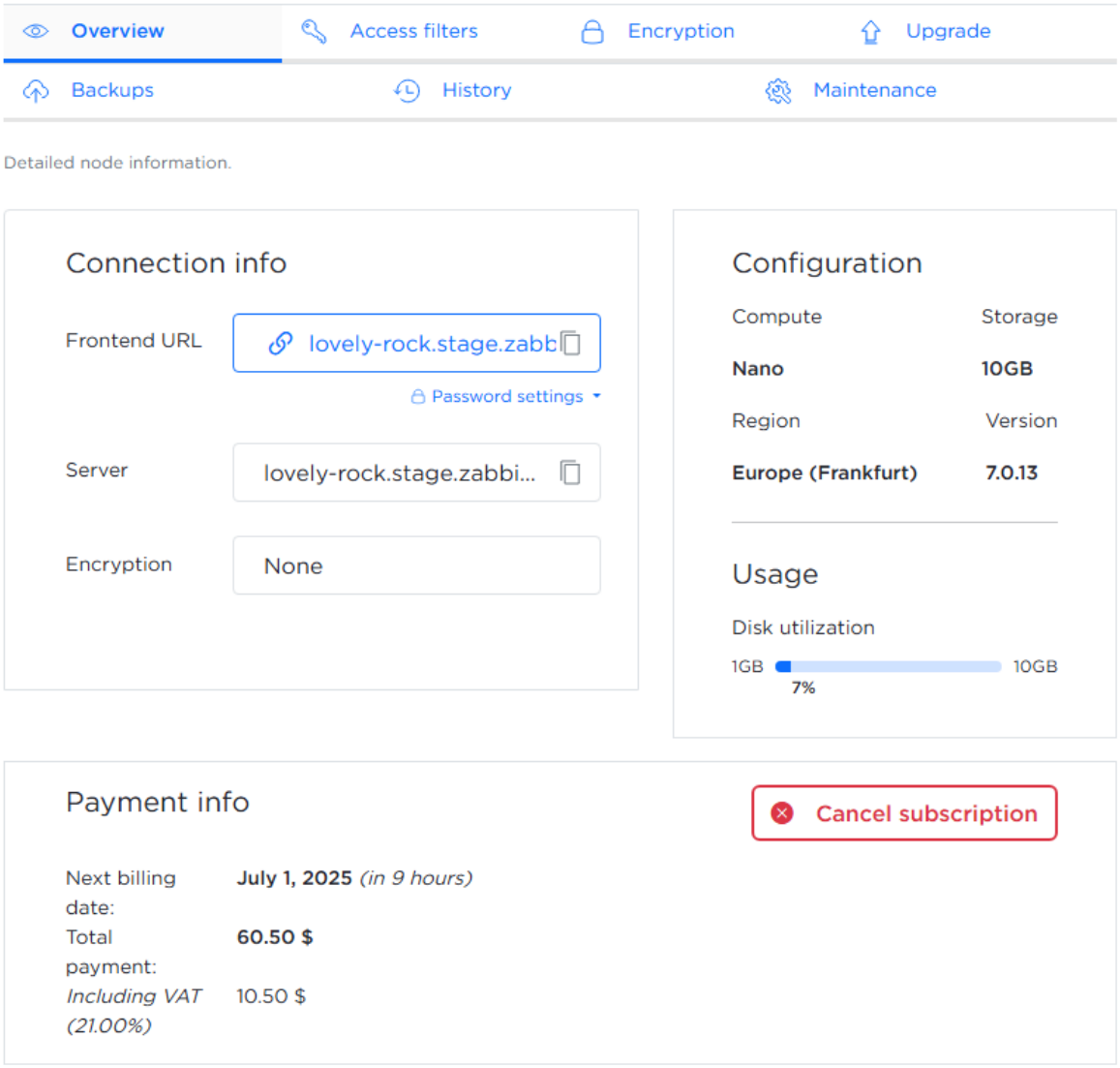


Note:
Niektóre karty mogą być niedostępne w zależności od typu węzła (serwer lub proxy), typu użytkownika wyświetlającego stronę oraz statusu węzła (wersja próbna/bez wersji próbnej).

Karta Przegląd

Karta *Przegląd* zawiera szczegółowe informacje o węźle. Wyświetlane informacje zależą od tego, czy węzeł jest serwerem, czy proxy.

Węzeł serwera Karta *Przegląd* węzła serwera zawiera następujące sekcje:



Informacje o połączeniu:

- Klikalny adres URL frontendu z opcją kopiowania do schowka
- Nazwa użytkownika
- Hasło węzła i jego ustawienia
- Nazwa hosta serwera z opcją kopiowania do schowka
- Adres IPv4 z opcją kopiowania do schowka
- Adres IPv6 z opcją kopiowania do schowka
- Bieżąca metoda szyfrowania

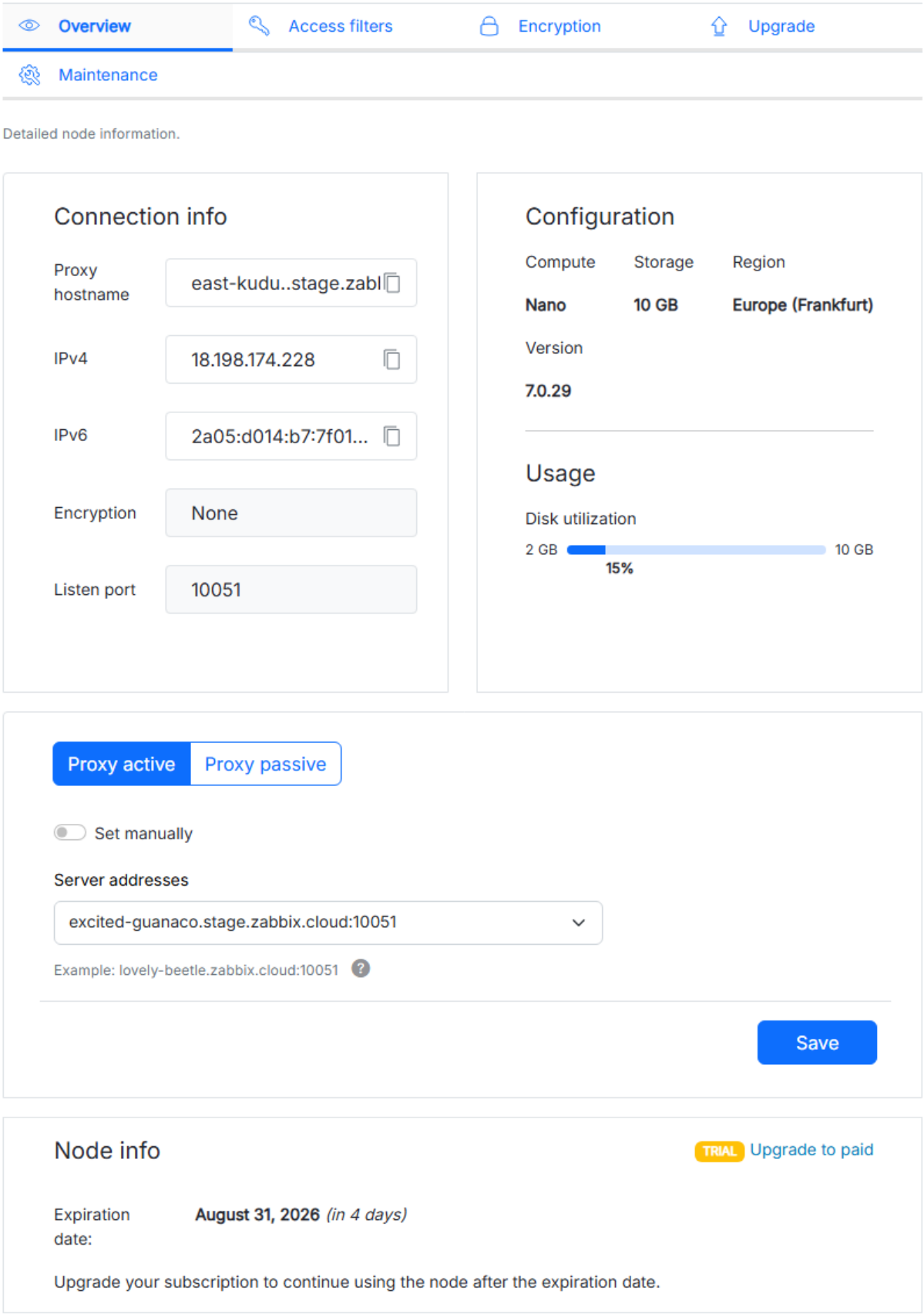
Konfiguracja i użycie:

- Poziom obliczeniowy, maksymalna ilość miejsca, region
- Wersja komponentów Zabbix
- Wykorzystanie dysku

Informacje o węźle/Informacje o płatności:

- Data wygaśnięcia (w przypadku węzłów testowych i przedpłaconych)
- Po zakończeniu inicjalizacji węzła wyświetlany jest przycisk zależny od typu węzła: - *Przejdź na płatny* - dostępny dla węzłów w ramach bezpłatnego okresu próbnego - *Anuluj subskrypcję* - dostępny dla płatnych węzłów z aktywną subskrypcją - *Ponownie zasubskrybuj plan płatny* - wyświetlany, jeśli subskrypcja została anulowana - W przypadku węzłów przedpłaconych nie jest wyświetlany żaden przycisk działania, ponieważ mają długi okres ważności i nie można ich anulować

Węzeł proxy Karta *Przegląd* węzła proxy zawiera następujące sekcje:



Informacje o połączeniu:

- Nazwa hosta proxy z opcją kopiowania do schowka
- Adres IPv4 z opcją kopiowania do schowka
- Adres IPv6 z opcją kopiowania do schowka
- Bieżąca metoda szyfrowania
- Bieżący port nasłuchiwania

Konfiguracja i użycie:

- Warstwa obliczeniowa, maksymalna ilość miejsca, region
- Wersja komponentów Zabbix
- Wykorzystanie dysku

Tryb proxy:

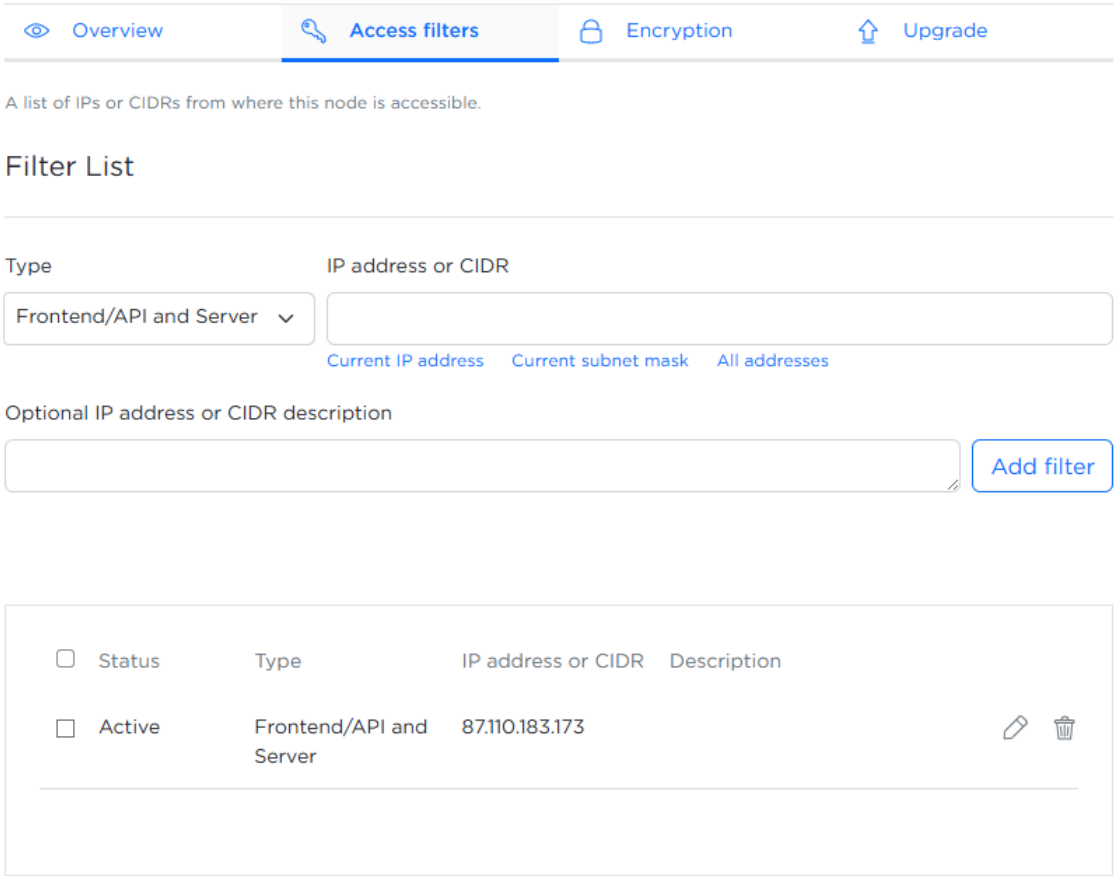
- Przełączanie między aktywnym i pasywnym trybem proxy
- Wybór adresu serwera z listy rozwijanej *Adresy serwera* lub włączenie opcji *Ustaw ręcznie*, aby ręcznie wprowadzić adres serwera

Informacje o węźle/*Informacje o płatności*:

- Data wygaśnięcia (w przypadku węzłów w okresie próbnym i opłaconych z góry)
- Po zakończeniu inicjalizacji węzła wyświetlany jest przycisk zależny od typu węzła: - *Przejdź na płatny* - dostępny dla węzłów w bezpłatnym okresie próbnym - *Anuluj subskrypcję* - dostępny dla płatnych węzłów z aktywną subskrypcją - *Ponownie zasubskrybuj wersję płatną* - wyświetlany po anulowaniu subskrypcji - W przypadku węzłów opłaconych z góry, które mają długi okres ważności i nie mogą zostać anulowane, przycisk akcji nie jest wyświetlany

Karta filtrów dostępu

Karta *Filtry dostępu* umożliwia ograniczenie dostępu do węzła Zabbix na podstawie adresu IP lub podsieci.



Aby skonfigurować filtr dostępu, wypełnij następujące pola:

- *Typ* - w przypadku węzłów serwera użyj listy rozwijanej, aby wybrać komponent Zabbix, którego dostęp ma zostać ograniczony: tylko frontend/API, tylko serwer lub oba. W przypadku węzłów proxy dostępna jest tylko opcja *Inbound IP*. Określa ona, które przychodzące żądania sieciowe mogą dotrzeć do proxy.
- *Adres IP lub CIDR* - wprowadź adres IP lub CIDR ręcznie albo wypełnij pole automatycznie, wybierając odpowiednią opcję.
- *Opcjonalny opis adresu IP lub CIDR* - dodaj opcjonalny opis każdego adresu IP lub podsieci.

Po wypełnieniu pól kliknij *Add filter*. Dodane filtry pojawią się w poniższej tabeli, która zawiera informacje o ich stanie, typie, adresie IP lub podsieci oraz opisie.

Warning:
Podczas inicjalizacji węzła automatycznie tworzony jest filtr dostępu dla używanego adresu IP. Jeśli później połączysz się z innego adresu IP lub innej sieci, musisz dodać ten adres tutaj, w przeciwnym razie dostęp zostanie zablokowany.

Możesz:

-  - edytować filtr.
-  - usunąć filtr lub wiele filtrów za pomocą pól wyboru.

Karta Encryption

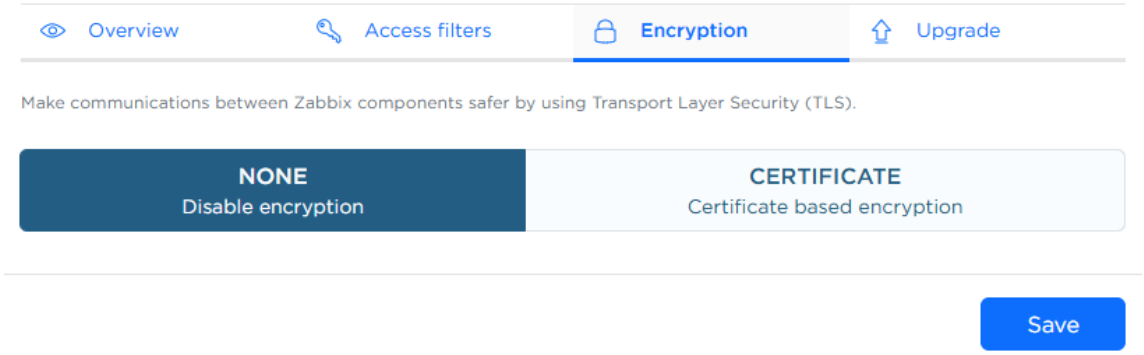
Karta *Encryption* umożliwia skonfigurowanie szyfrowania TLS na potrzeby komunikacji między komponentami Zabbix przez przesłanie wymaganych plików certyfikatów.

Aby włączyć bezpieczną komunikację przy użyciu certyfikatów, wybierz *CERTIFICATE* i prześlij z komputera następujące pliki:

- *Root CA certificates* - zaufane certyfikaty CA używane do weryfikowania certyfikatów peerów. Kliknij *Choose file* i wybierz plik zawierający zaufane certyfikaty Root CA (zwykle plik `.crt` lub `.pem`, zakodowany w formacie PEM).
- *Certificate chain* - reprezentuje łańcuch zaufania dla własnego certyfikatu. Kliknij *Choose file* i wybierz plik zakodowany w formacie PEM (np. `.crt` lub `.pem`), który zawiera własny certyfikat oraz ewentualne pośrednie certyfikaty CA.

- *Private key* - klucz prywatny odpowiadający własnemu certyfikatowi. Kliknij *Choose file* i wybierz plik `.key` zakodowany w formacie PEM.
- *Revoked certificates* (opcjonalnie) - jeśli jest utrzymywana lista unieważnionych certyfikatów (CRL), kliknij *Choose file* i wybierz plik CRL, aby określić certyfikaty, którym nie należy już ufać.

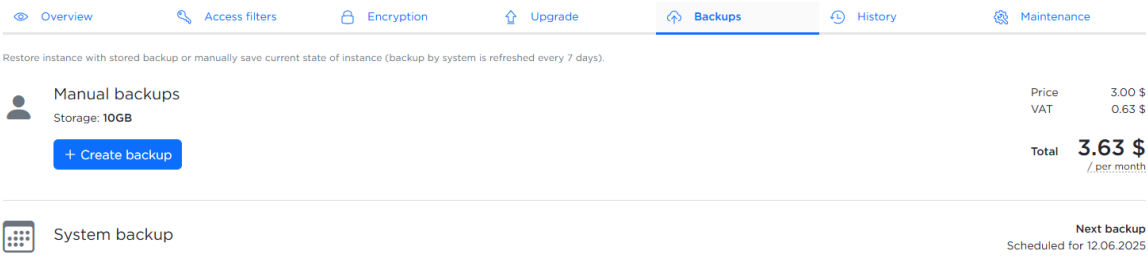
Po wybraniu plików kliknij *Save*. Pliki zostaną przesłane, zweryfikowane i zastosowane w celu skonfigurowania szyfrowania TLS między komponentami Zabbix.



Karta Kopie zapasowe

Karta *Kopie zapasowe* umożliwia wyświetlanie, tworzenie, przywracanie i usuwanie kopii zapasowych węzła Zabbix.

Instancję można przywrócić z dowolnej zapisanej kopii zapasowej lub w dowolnym momencie utworzyć nową kopię zapasową ręcznie.
Na karcie wyświetlana jest również data utworzenia następnej zaplanowanej kopii zapasowej.



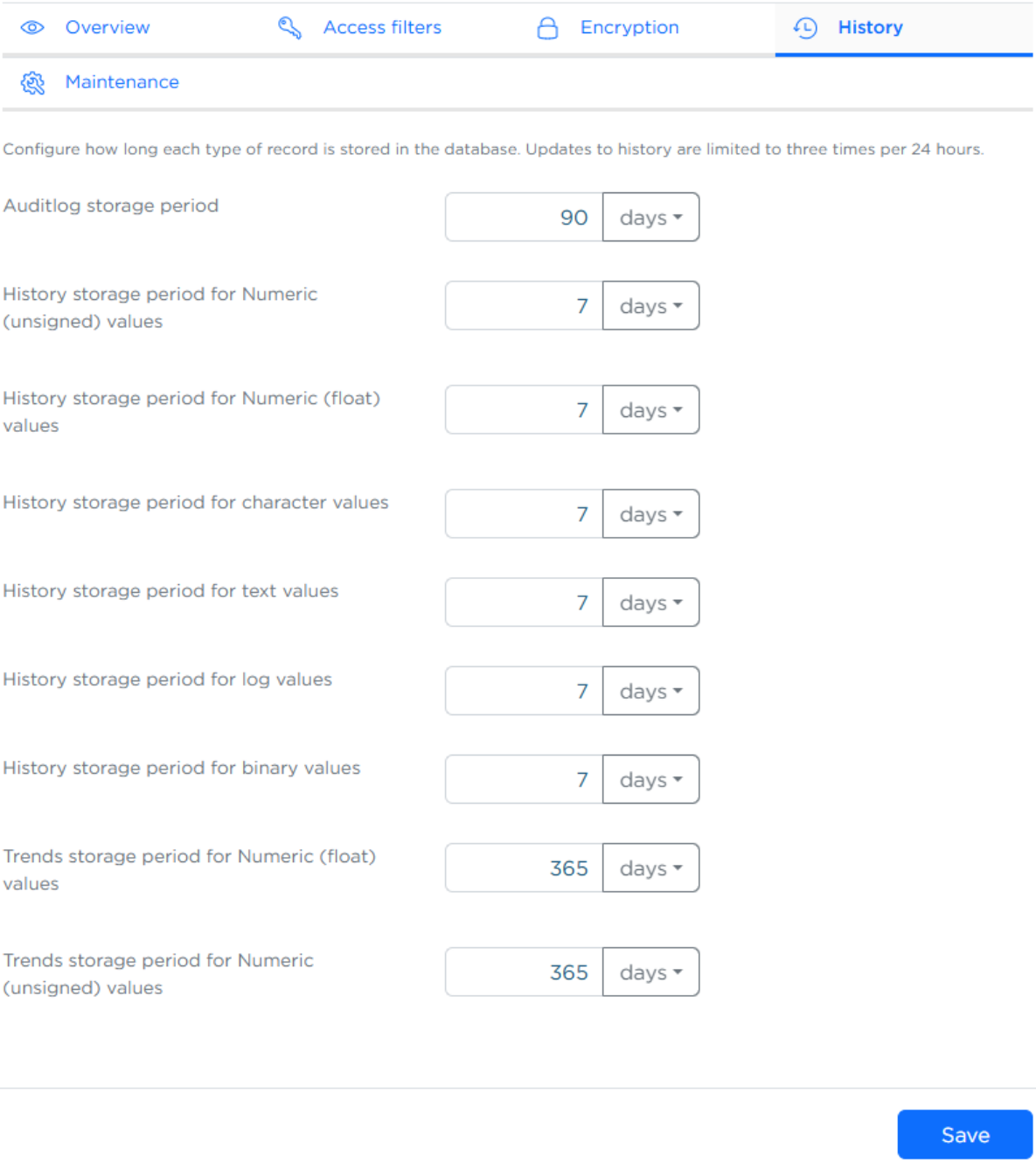
Aby utworzyć kopię zapasową:

1. Kliknij przycisk *Utwórz kopię zapasową*.
2. W wyskakującym oknie wprowadź opcjonalny opis.
3. Kliknij przycisk *Utwórz*, aby potwierdzić.

Warning:
Dodatkowe kopie zapasowe są rozliczane co miesiąc na podstawie wykorzystania przestrzeni dyskowej.

Karta historii

Karta *Historia* umożliwia skonfigurowanie czasu przechowywania różnych typów danych historycznych.



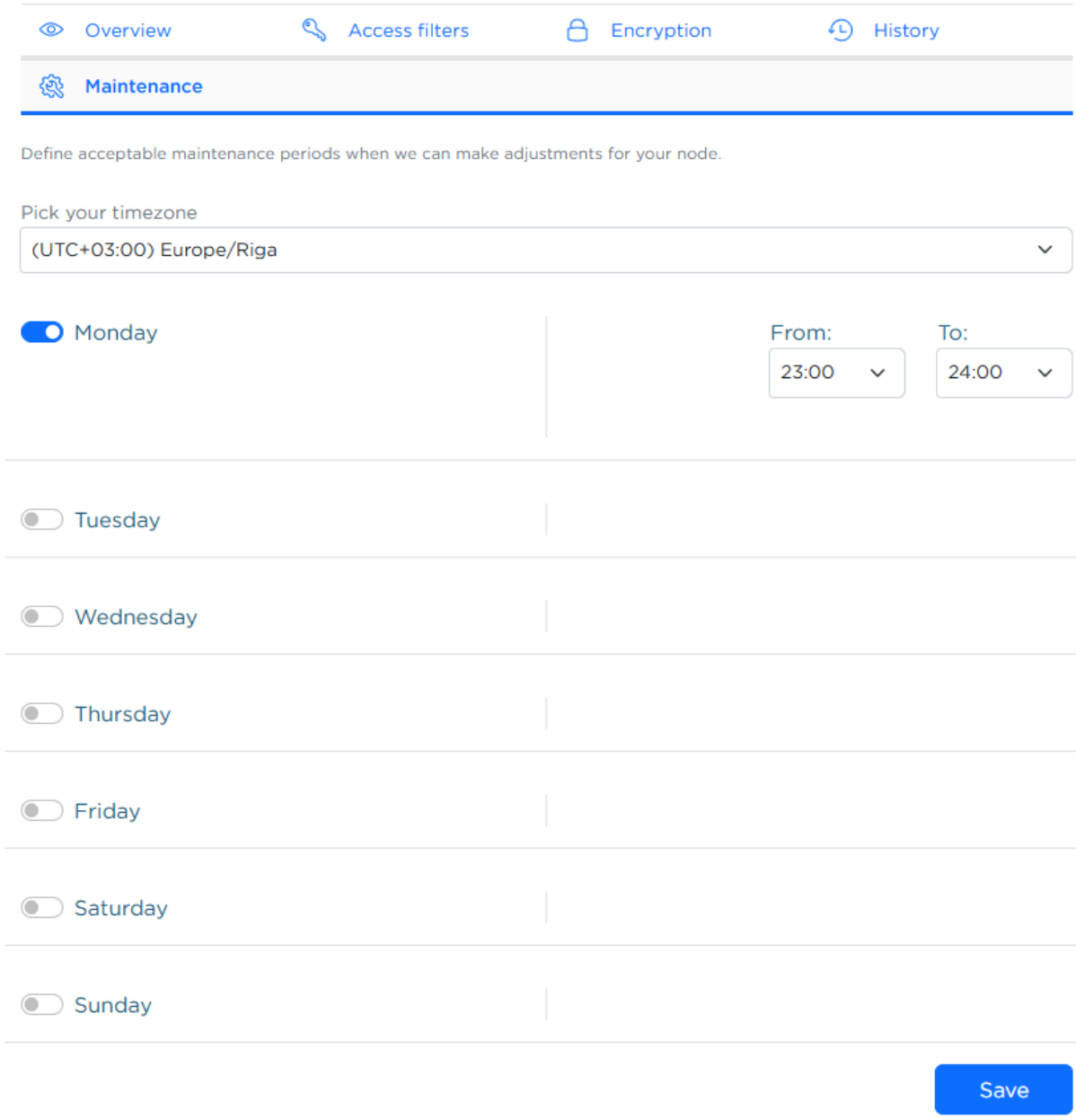
Możesz ustawić okres przechowywania dla każdego typu danych (np. liczbowych, tekstowych i dzienników) oraz wybrać jednostkę (dni, miesiące, lata). Zmiany można wprowadzać maksymalnie trzy razy w ciągu 24 godzin.

Po zaktualizowaniu wartości kliknij *Zapisz*, aby zastosować zmiany.

Warning:
Dłuższe okresy przechowywania mogą zwiększyć wykorzystanie miejsca na dysku i wpłynąć na wydajność. Zaleca się skonfigurowanie okresów przechowywania na podstawie rzeczywistych danych i potrzeb analitycznych.
 Zabbix Cloud używa partycjonowania bazy danych do zarządzania przechowywaniem danych zamiast tradycyjnego procesu housekeeping w Zabbix. Umożliwia to szybsze działanie i bardziej przewidywalne zarządzanie przechowywaniem danych.

Karta Maintenance

Karta *Maintenance* umożliwia zdefiniowanie dozwolonych okien konserwacji, w których można wykonywać aktualizacje lub dostosowania w Twoim węźle.

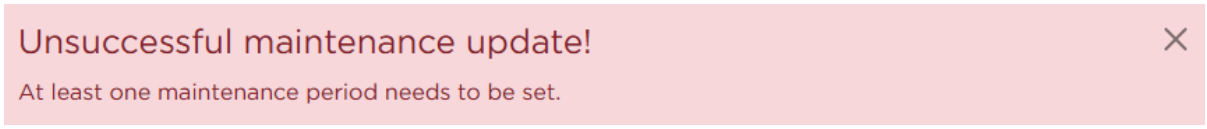


Należy:

- Wybrać strefę czasową z listy rozwijanej.
- Wybrać co najmniej jeden dzień i jedno godzinowe okno konserwacji.

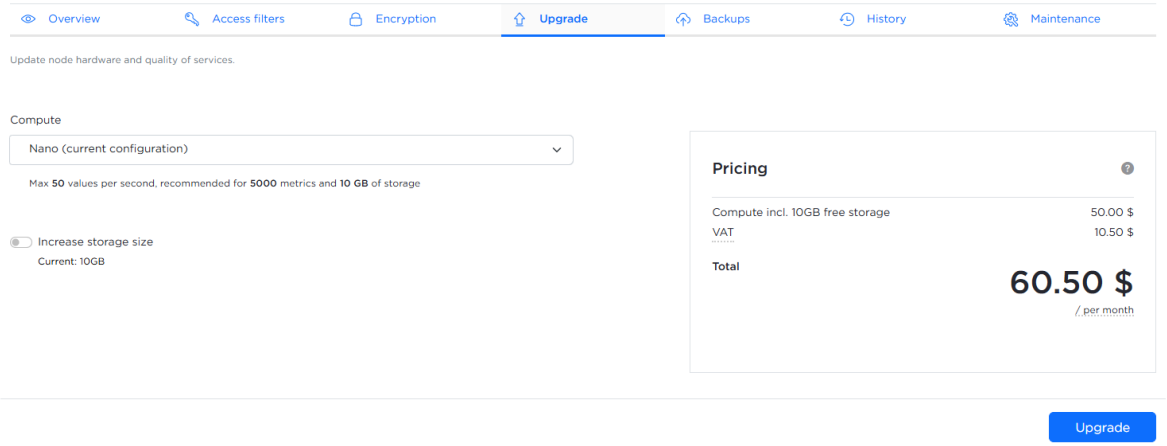
Kliknij *Save*, aby zastosować zmiany.

Jeśli nie zostanie wybrany dzień ani przedział czasowy, zostanie wyświetlony błąd.



Karta „Upgrade”

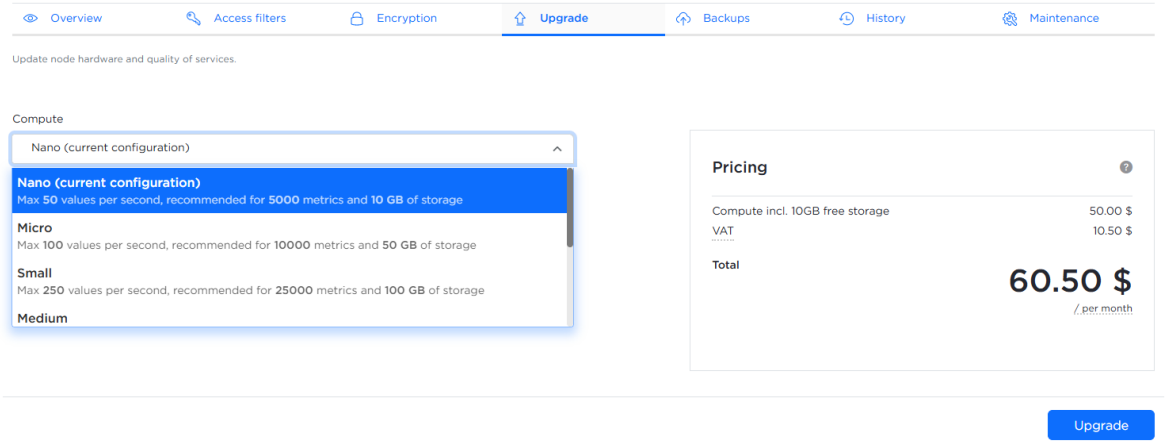
Karta *Upgrade* umożliwia skalowanie zasobów node’a lub zarządzanie jego subskrypcją.



Możesz zwiększyć poziom zasobów obliczeniowych lub rozmiar dysku. Pamiętaj, że rozmiar dysku można zwiększyć tylko raz na 24 godziny i nie można go zmniejszyć za pośrednictwem interfejsu użytkownika (w sprawie próśb o zmniejszenie rozmiaru skontaktuj się z pomocą techniczną Zabbix). Node'y w ramach bezpłatnego okresu próbnego można przekształcić w płatne subskrypcje.

Aby zwiększyć poziom zasobów obliczeniowych:

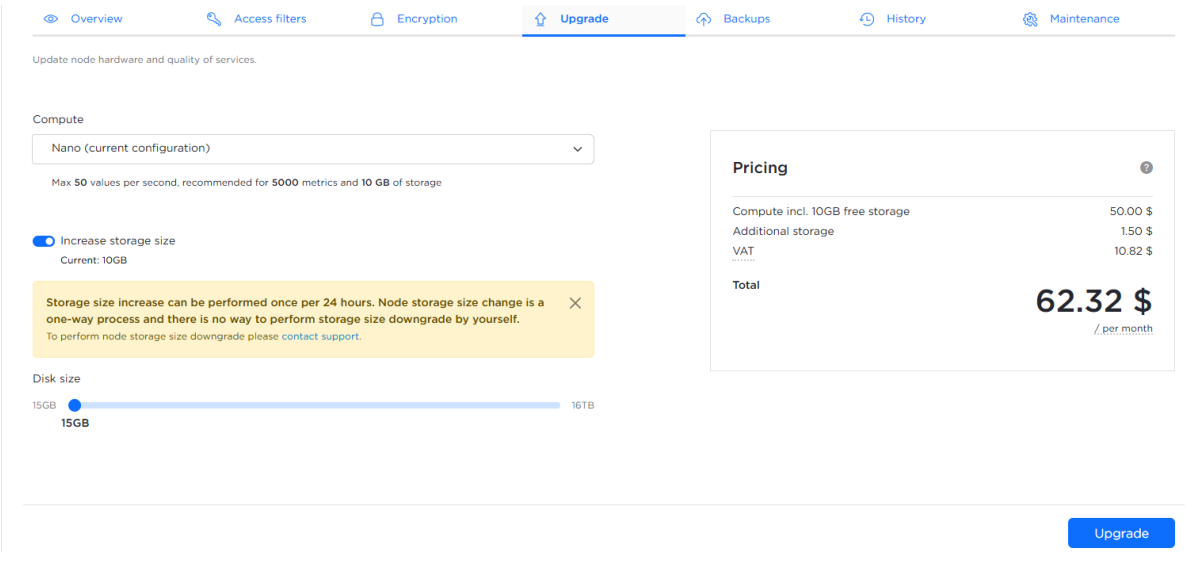
1. Kliknij listę rozwijaną *Compute*.



2. Wybierz żądany poziom konfiguracji.
3. Kliknij przycisk *Upgrade*.

Aby zwiększyć rozmiar dysku:

1. Kliknij przycisk radiowy *Increase storage size*.
2. Ustaw suwak *Disk size* na żądaną wartość.



3. Kliknij *Upgrade*.

Note:
Zaktualizowana łączna cena miesięczna jest wyświetlana w tabeli *Pricing* przed potwierdzeniem zmian.

Dodawanie użytkowników

Wprowadzenie

Poniższe przykłady opisują, jak dodawać użytkowników do organizacji i węzłów.

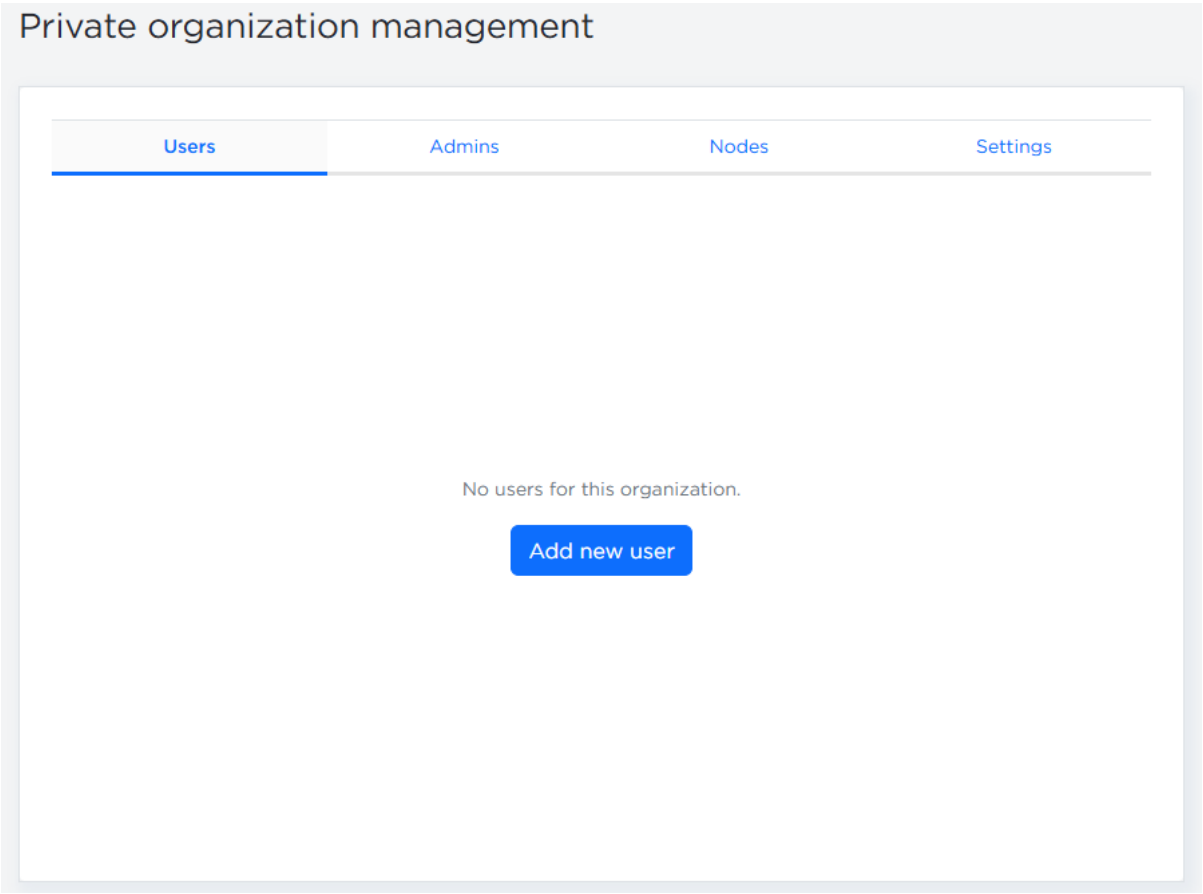
Note:
Kopiowanie, czyszczenie i resetowanie hasła frontend może być wykonywane wyłącznie przez właściciela.

Dodawanie użytkowników do organizacji

Podobnie jak użytkownicy w Zabbix, możesz tworzyć użytkowników w Zabbix Cloud i przypisywać im role, aby kontrolować, co mogą robić w organizacji i jej węzłach.

Aby dodać użytkowników do swojej organizacji:

1. Otwórz stronę *All organizations*.
2. Kliknij przycisk ustawień wybranej organizacji, aby przejść do jej menu zarządzania.



3. Na karcie *Users* lub *Admins* kliknij *Add new user*.
4. Wprowadź adres e-mail użytkownika i wybierz rolę: *Administrator* lub *User*.

Pamiętaj, że wybranie roli *User* dodaje użytkownika do organizacji, ale nie nadaje dostępu do żadnych węzłów.

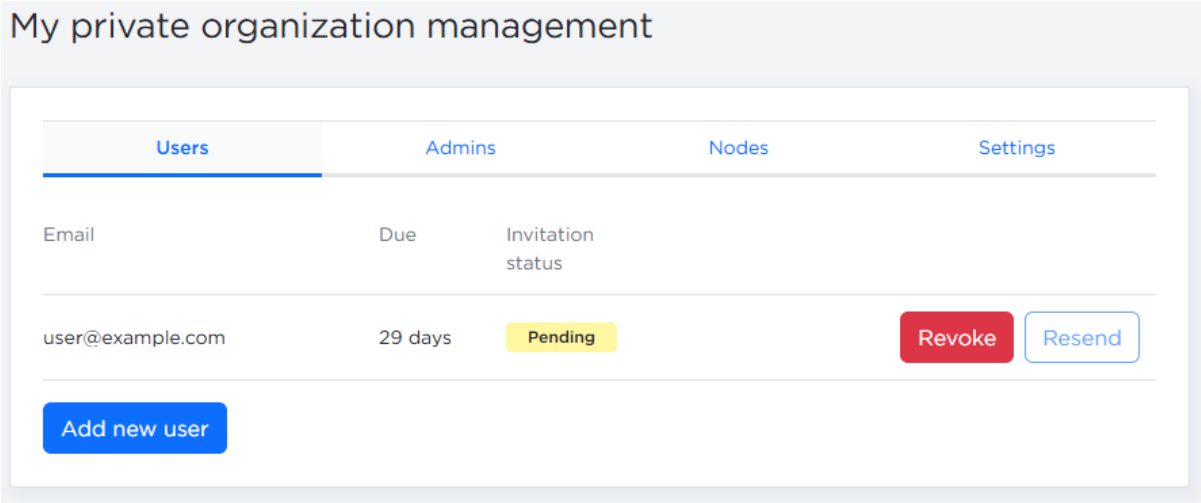
Każda rola zapewnia inny poziom dostępu:

- **User** — może uzyskiwać dostęp do przypisanych mu węzłów i wykonywać zadania konfiguracyjne, które nie generują opłat, na przykład:

  - Zarządzać filtrami dostępu
  - Konfigurować szyfrowanie
  - Ustawiać okresy przechowywania historii/trendów
  - Planować okresy konserwacji
- **Administrator** — ma wszystkie uprawnienia roli User i może również wykonywać działania, które mogą generować opłaty, na przykład:
  - Tworzyć i przywracać kopie zapasowe
  - Zwiększać lub zmniejszać zasoby węzła (warstwę obliczeniową i rozmiar dysku)
  - Anulować subskrypcje
- **Owner** — ma pełną kontrolę nad organizacją. Oprócz wszystkich uprawnień roli Administrator, Owner może:
  - Tworzyć i usuwać węzły
  - Dodawać lub usuwać użytkowników i administratorów
  - Zmieniać nazwę organizacji lub ją usuwać
  - Zarządzać danymi rozliczeniowymi

> **Attention:**
> Metoda płatności Owner jest obciążana za wszystkie usługi. Jeśli wymagane jest potwierdzenie bankowe, Owner musi autoryzować transakcję, nawet jeśli została zainicjowana przez Administratora.

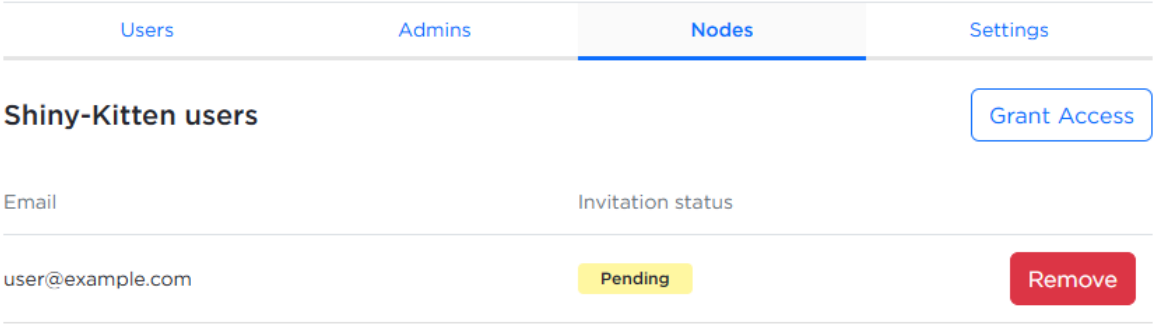
Zaproszony użytkownik zostanie dodany ze statusem *Pending*.
Możesz cofnąć zaproszenie lub wysłać je ponownie (ponowne wysłanie jest dostępne raz na 10 minut — przycisk będzie wyłączony po osiągnięciu tego limitu).



Dodawanie użytkowników do węzłów

Aby dodać użytkownika do węzła:

1. Przejdź do zakładki *Nodes*.
2. Kliknij *Grant Access*.
3. Wybierz z listy rozwijanej użytkownika dodanego do organizacji.
4. Kliknij *Grant access*.



Alternatywnie możesz nadać dostęp z okna *Node configuration*:

1. Kliknij przycisk ustawień  .

2. Wybierz *Add user*.
3. Wybierz użytkownika i kliknij *Grant access*.

Zaproszeni użytkownicy muszą zaakceptować zaproszenie, tworząc konto Zabbix Cloud przy użyciu tego samego adresu e-mail, na który wysłano zaproszenie. Jeśli użytkownik ma już konto Zabbix Cloud przypisane do tego adresu e-mail, musi się zalogować i potwierdzić zaproszenie. Zaproszenia są powiązane z konkretnym adresem e-mail i nie mogą zostać zaakceptowane z innego konta.

Kluczowe różnice w Zabbix Cloud

Wprowadzenie

Zabbix Cloud to platforma chmurowa umożliwiająca użytkownikom uruchamianie samodzielnych instancji Zabbix w chmurze (węzłów chmurowych).

W porównaniu ze standardowym wdrożeniem Zabbix lokalnie:

- Nie jest wymagana instalacja ani konserwacja.
- Węzły chmurowe są w pełni zarządzane przez Zabbix.
- Węzły chmurowe mogą zostać uruchomione w ciągu kilku minut.

Usługa ta jest oferowana w modelu pay-as-you-go (dostępny jest **bezpłatny okres próbny**).

Tabela porównawcza Chociaż większość funkcji pokrywa się między lokalnymi instalacjami Zabbix a węzłami w chmurze, występują również różnice. Poniżej przedstawiono podsumowanie w formie tabeli porównawczej. W kolejnych sekcjach szczegółowo opisano wybrane elementy.

Funkcja / aspekt	Zabbix Cloud	Zabbix lokalny
Cykl wydawniczy	Zgodny z wydaniami LTS	Samodzielnie wybierasz wersję Zabbix (stabilną, beta lub niestandardową)
Łatwość użycia	Gotowa do użycia platforma, w pełni zarządzana przez Zabbix	Wymaga instalacji, konfiguracji i konserwacji
Usuwanie luk w zabezpieczeniach	Automatyczne aktualizacje systemu operacyjnego i Zabbix	Ręczne aktualizacje wykonywane przez użytkownika
Certyfikat HTTPS	Ważny certyfikat dostępny od razu (bez Let's Encrypt)	Domyślnie samopodpisany (można skonfigurować Let's Encrypt lub inny urząd certyfikacji)
Wydajność bazy danych (INSERT/UPDATE/SELECT)	Automatyczne partycjonowanie zapewniające maksymalną szybkość; brak długo trwających operacji DELETE	Ręczne partycjonowanie i czyszczenie (możliwe długotrwałe transakcje DELETE)
Dostrajanie wydajności	Wstępnie dostrojone przez zespół Zabbix Cloud (bez dzienników zmian)	Pełny wgląd w parametry dostrajania i kontrola nad nimi
Kontrola zapory sieciowej	Zarządzana przez zespół Zabbix Cloud; porty GUI/trapper nie są publicznie dostępne	Pełna kontrola nad regułami zapory i udostępnianiem portów
Ruch wychodzący na porcie 25	Domyślnie blokowany w celu zapobiegania spamowi i nadużyciom; zamiast tego użyj portu SMTP 587 lub 465	Nie jest ograniczany przez Zabbix; zależy od Twojej sieci i dostawcy hostingu
Izolacja i ochrona bazy danych	Wysoki poziom izolacji, brak bezpośredniego dostępu do bazy danych/SSH	Zależy od Twojej sieci i zabezpieczeń hosta
Wysoka dostępność (HA)	Działa na wysoce dostępnym środowisku chmurowym, proxy Zabbix można używać w celu zwiększenia odporności gromadzenia danych	HA zarządzane przez użytkownika
Pułapki SNMP	Tylko za pośrednictwem dedykowanego proxy (brak odbiornika pułapek HA/równoważonego obciążeniem)	Bezpośrednio lub za pośrednictwem proxy, możliwe HA/równoważenie obciążenia
Odpytywanie SNMP	Trudne bez proxy (każde urządzenie wymaga NAT/niestandardowego portu)	Natywne odpytywanie SNMP; proxy opcjonalne

Funkcja / aspekt	Zabbix Cloud	Zabbix lokalny
Ustawienia przechowywania danych	Historia/trendy/audyt tylko za pośrednictwem interfejsu użytkownika (brak API lub pliku konfiguracyjnego)	Konfigurowalne w pliku konfiguracyjnym serwera Zabbix lub za pośrednictwem API
Skrypty niestandardowe	`AlertScriptsPath`, `ExternalScripts`, frontend i moduły społecznościowe nie są obsługiwane	W pełni obsługiwane (ścieżka do skryptów, moduły, integracje)
Monitorowanie ODBC	Tylko PostgreSQL (sterownik `{postgresql}`); wtyczka MariaDB jest dostępna jako symbol zastępczy	ODBC dla PostgreSQL, MySQL, Oracle itd. - w pełni konfigurowalne
Ograniczanie wywołań ODBC	Niemożliwe (dostępne jest tylko `StartODBCPollers=1`); ciężkie zapytania synchronizacyjne blokują inne zadania	W pełni konfigurowalna liczba pollerów i harmonogram
Przesyłanie certyfikatu SAML	Nieobsługiwane	Obsługiwane za pośrednictwem interfejsu użytkownika lub API
Raporty cykliczne	Należy utworzyć własny nośnik Email (nośnik skryptowy nie jest obsługiwany)	Domyślnie obsługiwane są zarówno nośniki skryptowe, jak i Email
Interfejs hosta dla aktywnych kontroli	Adres IP przydzielony przez platformę (tworzony automatycznie)	Interfejsy zarządzane przez Ciebie; adresy IP pozostają pod Twoją kontrolą

Różnice funkcjonalne Pułapki SNMP

Pułapki SNMP są obsługiwane wyłącznie przez dedykowany proxy Zabbix. Jeśli wymagane jest monitorowanie pułapek SNMP, nie można używać automatycznego równoważenia obciążenia ani wysokiej dostępności dla proxy, ponieważ pułapki SNMP muszą być wysyłane na pojedynczy adres IP.

Odpytywanie SNMP

Bez proxy odpytywanie SNMP wymaga, aby każde urządzenie było udostępnione przez NAT i niestandardowe porty.

Ustawienia retencji

Okresy retencji historii, trendów i dziennika audytu można konfigurować w chmurze wyłącznie za pośrednictwem interfejsu webowego. Tych ustawień nie można konfigurować za pomocą `zabbix_server.conf` ani przez API.

Ręczne nadpisywanie historii dla poszczególnych pozycji nie jest obsługiwane (partycjonowanie jest kontrolowane globalnie). Adresy URL węzłów chmurowych nie akceptują tych samych parametrów zapytania co instalacje lokalne.

Własne skrypty

Następujące typy własnych skryptów nie są obsługiwane przez Zabbix Cloud:

- Skrypty alertów (`AlertScriptsPath`)
- Skrypty zewnętrzne (`ExternalScripts`)
- Skrypty frontend na serwerze Zabbix

Nie można instalować modułów frontend opracowanych przez społeczność.

Monitorowanie ODBC

Zabbix Cloud obsługuje monitorowanie ODBC tylko dla PostgreSQL. Użyj oficjalnego szablonu ODBC Zabbix i zdefiniuj connection string w szablonie, używając:

```
Driver={postgresql}
```

Wtyczka MariaDB jest również zainstalowana, ale obecnie nie działa. Jeśli jest używana, zdefiniuj:

```
Driver={mysql}
```

Attention:
Nie jest znana żadna działająca konfiguracja dla Oracle MySQL 8.0 w Zabbix Cloud. Chociaż proste zapytania (takie jak `SELECT 1`) mogą się powieść, bardziej złożone zapytania zwracają `SQL_ERROR`.

 Nie ma możliwości ograniczenia wywołań ODBC (`StartODBCPollers=1` only). Intensywne synchroniczne raportowanie może wpływać na wydajność, a jednocześnie może być uruchomione tylko pojedyncze `SELECT`.

Dostęp do infrastruktury

Zabbix Cloud nie zapewnia dostępu SSH do bazowych węzłów ani nie zezwala na bezpośrednie połączenia z bazą danych (na przykład do portu 3306). Wszystkie działania związane z konfiguracją, monitorowaniem i rozwiązywaniem problemów muszą być

wykonywane za pośrednictwem interfejsu użytkownika chmury lub API, co zapewnia, że warstwy systemu operacyjnego i bazy danych pozostają odizolowane i zabezpieczone.

Ruch wychodzący na porcie 25

Aby zapobiegać spamowi i nadużyciom, połączenia wychodzące z węzłów Zabbix Cloud do portu TCP 25 są domyślnie blokowane. Dotyczy to:

- typów nośników Email, które wysyłają wiadomości za pośrednictwem serwera SMTP na porcie 25. Ponieważ 25 to domyślny *port serwera SMTP* dla nowego typu nośnika Email, zmień go na 587 (z ustawieniem *Zabezpieczenia połączenia* na STARTTLS) lub 465 (SSL/TLS), zależnie od tego, co obsługuje Twój serwer SMTP. Dostawca poczty e-mail *Office365 relay* nie może być używany, ponieważ łączy się z punktem końcowym przekaźnika firmy Microsoft na porcie 25.
- Raporty zaplanowane, które wymagają typu nośnika Email.
- Sprawdzenia wykonywane przez sam węzeł chmurowy na porcie 25, takie jak `net.tcp.service[smtp]` proste sprawdzenia lub sprawdzenia SMTP w regułach wykrywania sieci. Zgłaszają one usługę SMTP jako niedostępną, nawet gdy działa; zamiast tego wykonuj takie sprawdzenia za pośrednictwem proxy Zabbix lub agenta Zabbix w swojej sieci.

Domyślny typ nośnika *Cloud Email* nie jest objęty tym ograniczeniem.

Interfejs aktywnego sprawdzania

Aktywne sprawdzenia w chmurze automatycznie tworzą interfejs hosta z adresem IP niezwiązanym z Twoją siecią lokalną. Domyślnie ten adres IP może być zewnętrzny. Aby zachować spójność, możesz ręcznie usunąć lub dostosować interfejs hosta po jego utworzeniu.

Przesyłanie certyfikatu SAML

Uwierzytelnianie SAML nie jest obsługiwane, ponieważ niewielu dostawców akceptuje niezaszyfrowane lub niepodpisane komunikaty SAML.

Raporty zaplanowane

Domyślny typ nośnika *Cloud Email* korzysta z transportu skryptowego i nie może być używany do raportów zaplanowanych. Aby wysyłać raporty pocztą elektroniczną:

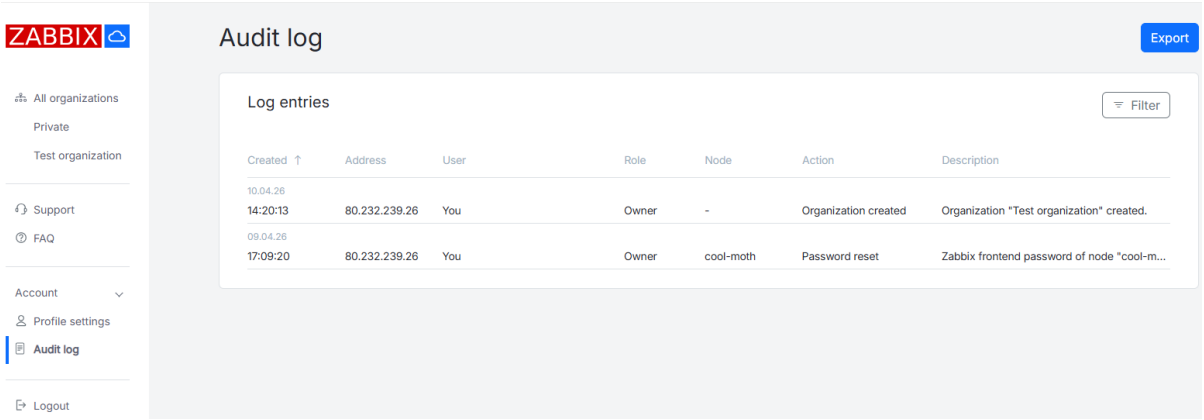
1. Utwórz nowy typ nośnika Email.
2. Przypisz nośnik e-mail do swojego użytkownika w sekcji *Ustawienia użytkownika* > *Nośniki*.
3. Wybierz ten typ nośnika podczas konfigurowania raportów zaplanowanych.

Attention:
Nie używaj portu 25 jako *portu serwera SMTP* dla tego typu nośnika; zobacz Ruch wychodzący na porcie 25.

Dziennik audytu

Przegląd W Zabbix Cloud dziennik audytu rejestruje aktywność użytkowników i systemu.

Po zalogowaniu się do konta Zabbix Cloud przejdź do *Account* > *Audit log*, aby wyświetlić wpisy dziennika audytu. Dostępne dla użytkownika wpisy dziennika audytu zależą od roli organizacyjnej użytkownika.



Wyświetlanie rekordów dziennika audytu Rekordy dziennika audytu są wyświetlane w tabeli.

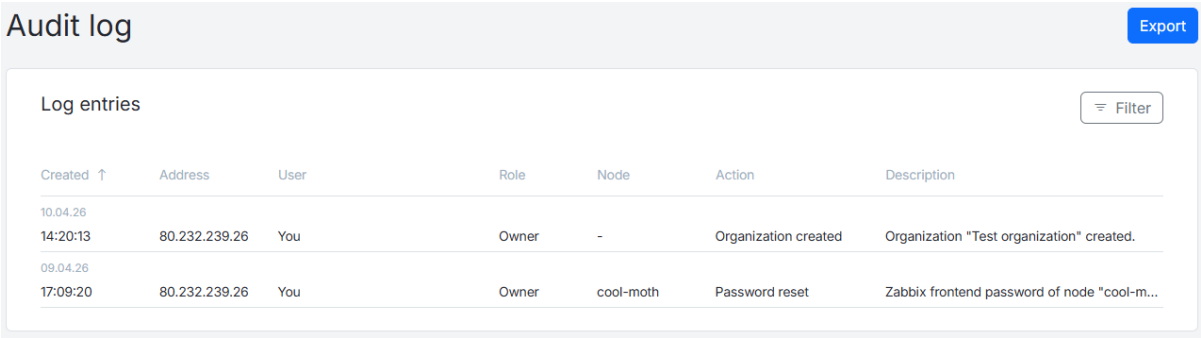


Tabela zawiera następujące informacje:

Column	Description
Created	Data i godzina utworzenia wpisu dziennika. Wyświetlany czas odpowiada strefie czasowej użytkownika wybranej w rozwijanym polu *User timezone* w *Account > Profile settings*.
User	Nazwa użytkownika organizacji, który wykonał akcję.
Role	Rola użytkownika, który wykonał akcję.
Node	Nazwa węzła powiązanego z akcją.
Action	Nazwa wykonanej akcji.
Description	Opis wykonanej akcji.

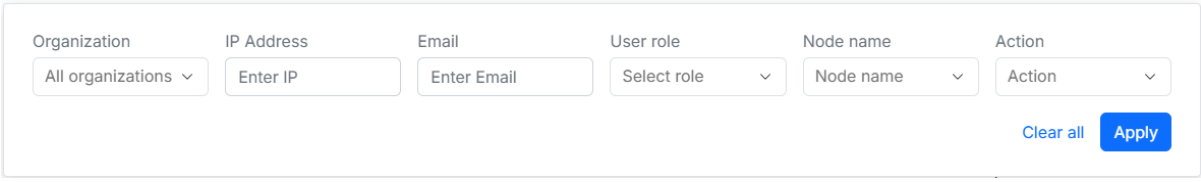
Nazwy kolumn można kliknąć, aby sortować rekordy rosnąco lub malejąco.

Przefiltrowane rekordy dziennika audytu można eksportować w formacie `.csv` za pomocą przycisku *Export*.

Korzystanie z filtrów

Użyj przycisku *Filter*, aby otworzyć filtr.

Aby wyświetlić odfiltrowane wyniki, po wybraniu co najmniej jednego parametru filtrowania kliknij *Apply*. Aby wyczyścić filtr, kliknij *Clear all*.



Rekordy można filtrować według jednego lub większej liczby parametrów. Dostępne są następujące parametry filtrowania:

Parametr	Opis
Organization	Wybierz wszystkie organizacje lub określoną organizację.
IP Address	Określ adres IP.
Email	Określ adres e-mail przypisany do użytkownika.
User role	Wybierz wszystkie role lub określoną rolę: Owner, Admin lub User.
Node name	Wybierz wszystkie węzły lub określony węzeł.
Action	Wybierz określony typ działania.

Dziennik audytu śledzi następujące działania:

- *Utworzenie węzła*
- *Usunięcie węzła*
- *Uaktualnienie węzła*
- *Aktualizacja wersji węzła*
- *Resetowanie hasła*
- *Aktualizacja ustawień konserwacji*
- *Aktualizacja ustawień historii*
- *Przedłużenie daty wygaśnięcia węzła*
- *Aktualizacja konfiguracji proxy*

- *Utworzenie kopii zapasowej*
- *Usunięcie kopii zapasowej*
- *Przywrócenie kopii zapasowej*
- *Aktualizacja kopii zapasowej*
- *Dodanie filtra dostępu*
- *Aktualizacja filtra dostępu*
- *Usunięcie filtra dostępu*
- *Włączenie szyfrowania*
- *Wyłączenie szyfrowania*
- *Anulowanie subskrypcji*
- *Odnowienie subskrypcji*
- *Nieudana płatność za subskrypcję*
- *Płatność wymaga potwierdzenia*
- *Utworzenie organizacji*
- *Usunięcie organizacji*
- *Zaproszenie administratora*
- *Zaproszenie użytkownika*
- *Dołączenie administratora*
- *Dołączenie użytkownika*
- *Usunięcie administratora z organizacji*
- *Usunięcie użytkownika z organizacji*
- *Dodanie węzła do organizacji*
- *Usunięcie węzła z organizacji*
- *Przyznanie dostępu do węzła*
- *Odebranie dostępu do węzła*
- *Aktualizacja ustawień użytkownika*
- *Utworzenie płatności*
- *Zrealizowanie płatności*
- *Nieudana płatność*
- *Naliczono opłatę VAT*
- *Nieudana walidacja VAT*
- *Aktualizacja informacji rozliczeniowych*
- *Aktualizacja metody płatności*
- *Usunięcie metody płatności*
- *Utworzenie węzła przedpłaconego*
- *Aktualizacja węzła przedpłaconego*
- *Rozpoczęcie incydentu*
- *Zakończenie incydentu*

Konfigurowanie proxy Zabbix Cloud

Przegląd Ta strona opisuje sposób konfigurowania proxy Zabbix wdrożonego w Zabbix Cloud w celu zbierania danych monitorowania z hostów w zdalnym środowisku.

Procedura jest przeznaczona przede wszystkim dla serwera Zabbix i proxy utworzonych w Zabbix Cloud.

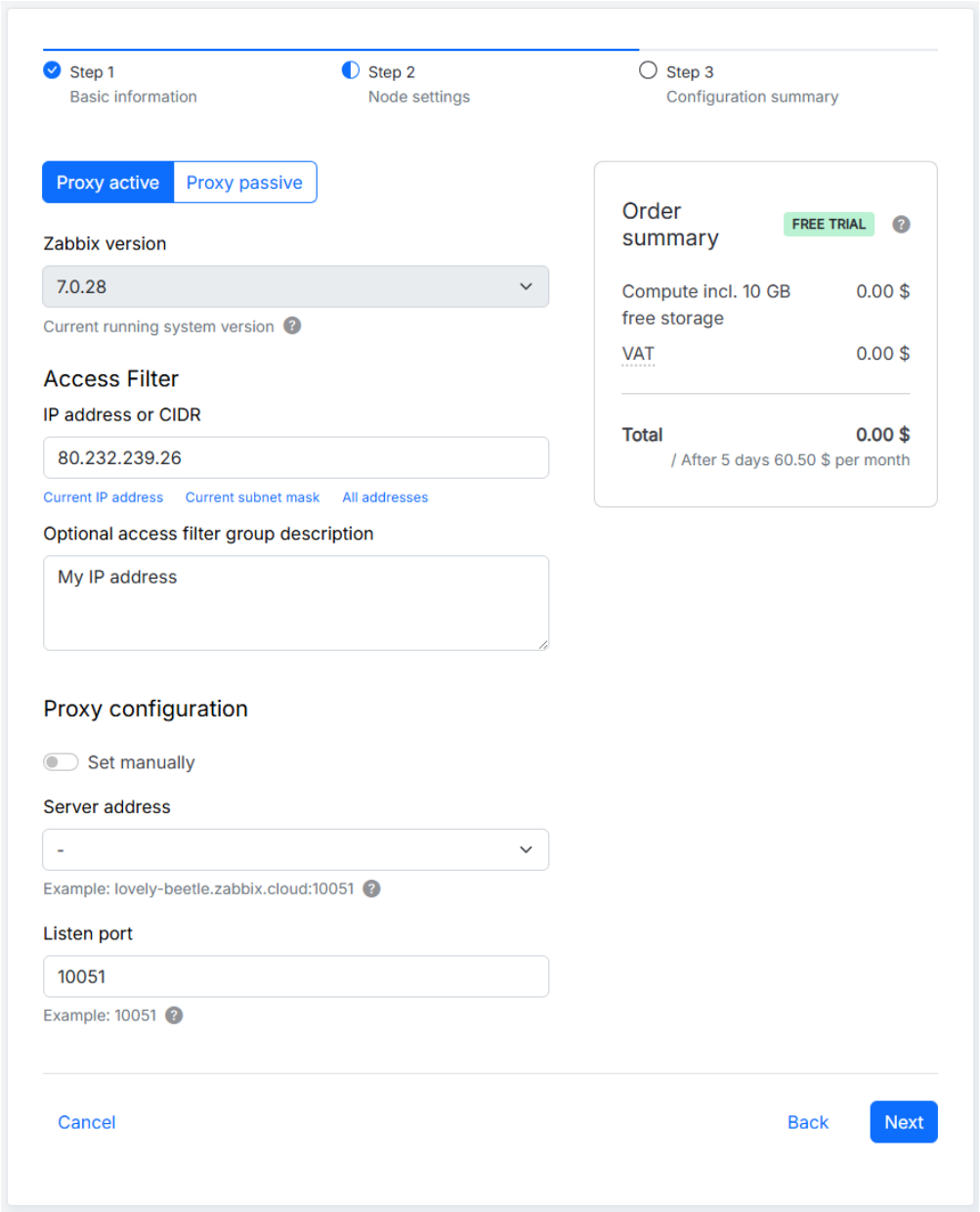
Procedura opisuje również konfigurację wymaganą, gdy serwer Zabbix nie jest hostowany w Zabbix Cloud.

Wymagania wstępne Serwer Zabbix jest zainstalowany, skonfigurowany i uruchomiony lub w Zabbix Cloud działa węzeł serwera Zabbix.

Utwórz węzeł proxy Utwórz węzeł proxy Zabbix Cloud.

Podczas tworzenia węzła:

1. Wybierz *Proxy*.
2. Wybierz *Proxy active* lub *Proxy passive* w zależności od tego, jak proxy powinien komunikować się z serwerem Zabbix.
3. Skonfiguruj adres IP węzła lub filtry dostępu CIDR (można to zrobić po utworzeniu węzła).
4. Określ serwer Zabbix, z którym proxy będzie się łączyć.



5. Zakończ proces tworzenia węzła.

Więcej informacji na temat tworzenia węzła proxy zawiera sekcja Wdróż Zabbix w chmurze.

Konfigurowanie filtrów dostępu Zabbix Cloud używa filtrów dostępu do kontrolowania, które adresy IP lub podsieci mogą łączyć się z węzłem chmurowym.

Podczas inicjalizacji węzła automatycznie tworzony jest filtr dostępu dla adresu IP używanego do uzyskania dostępu do węzła. Nie musi on obejmować adresów monitorowanych hostów ani innych komponentów Zabbix. Dodaj wymagane adresy ręcznie.

Więcej informacji można znaleźć w sekcji Konfiguracja węzła — filtry dostępu.

Węzeł proxy

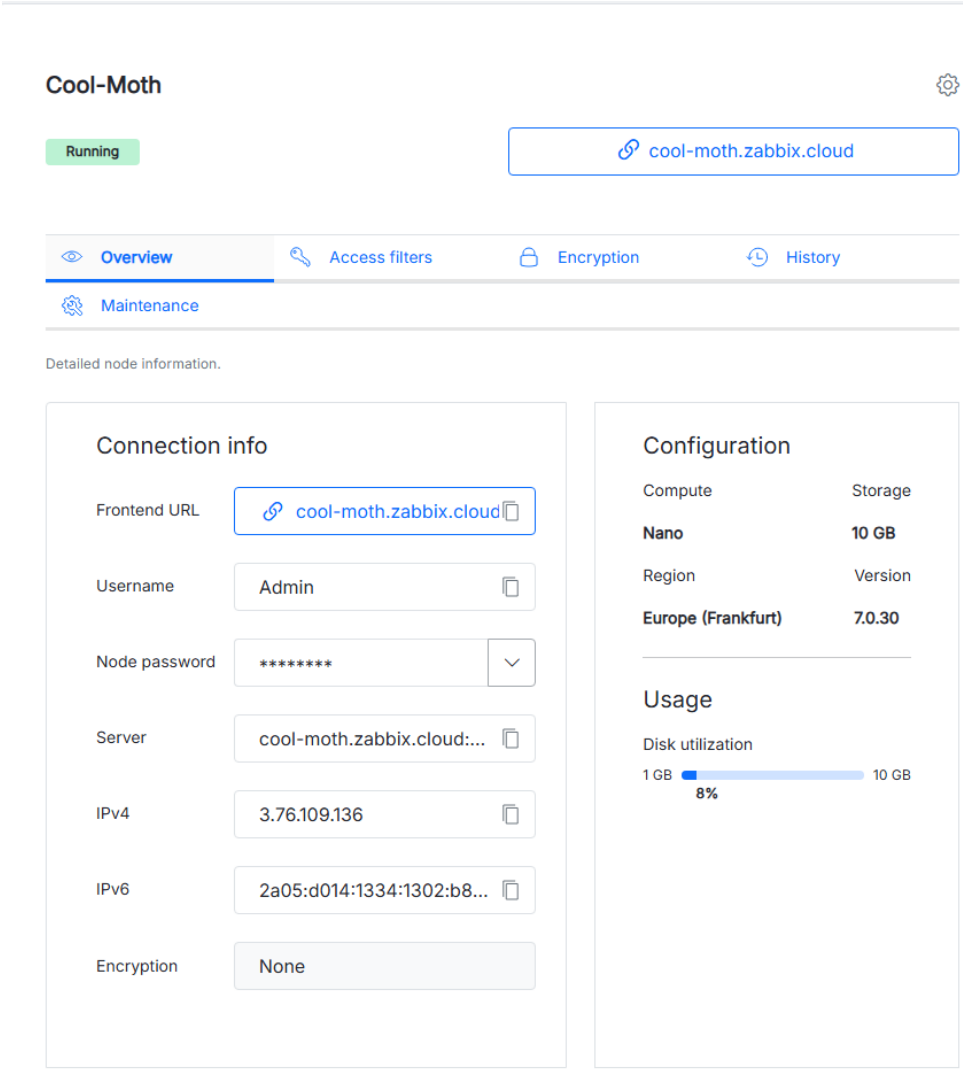
Dodaj adresy IP lub podsieci, z których oczekiwane są połączenia z proxy.

Jeśli monitorowane hosty używają aktywnych kontroli agentów, agenty inicjują połączenia z proxy, dlatego adres IP lub podsieć monitorowanych hostów musi być dozwolona przez filtry dostępu węzła proxy.

Jeśli jako tryb proxy ustawiono *Proxy Passive*, serwer Zabbix łączy się z proxy, dlatego adres IP lub podsieć serwera Zabbix musi być dozwolona przez filtry dostępu węzła proxy.

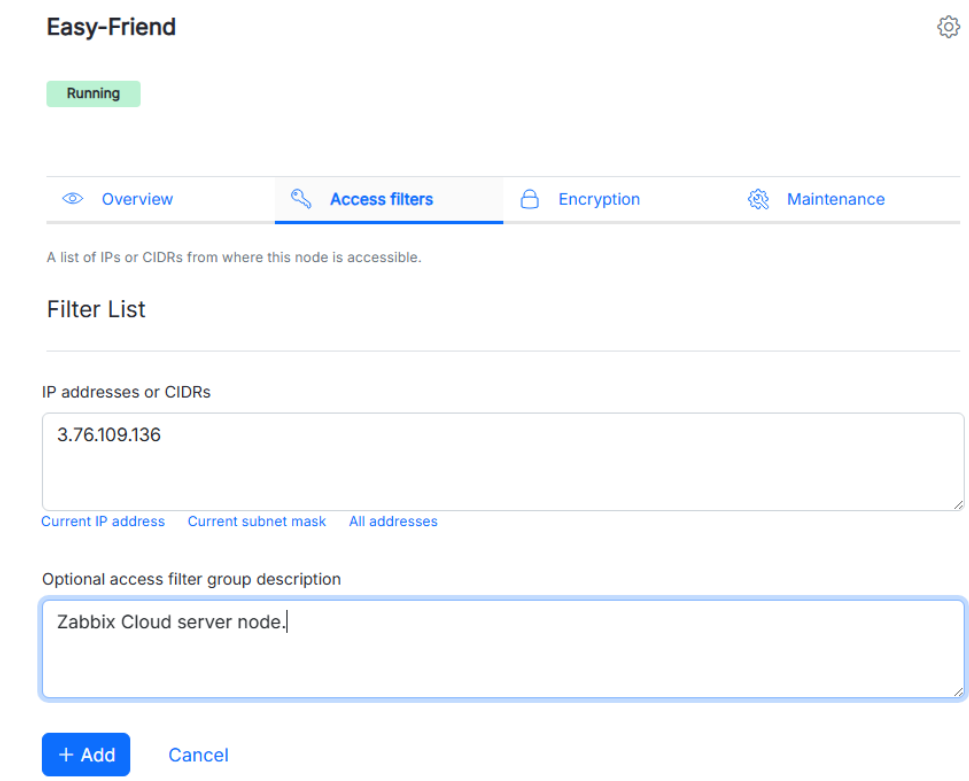
Jeśli wybrano *Proxy passive* i używany jest węzeł serwera:

1. Przejdź do karty *Overview* węzła serwera i skopiuj adres IPv4 lub IPv6 węzła.

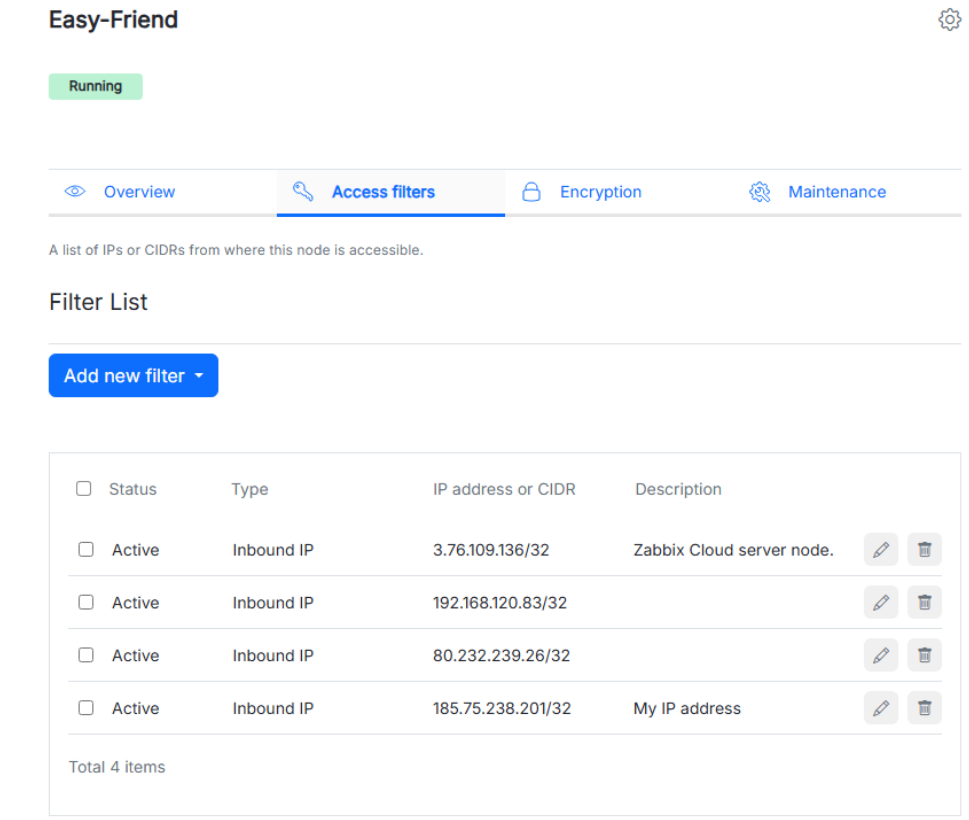


2. Przejdź do karty *Access filters* węzła proxy, kliknij *Add new filter* > *Add*.

3. Wklej skopiowany adres w polu *IP addresses or CIDRs*, a następnie dodaj opis w polu *Optional access filter group description*.



4. Kliknij *Add*.



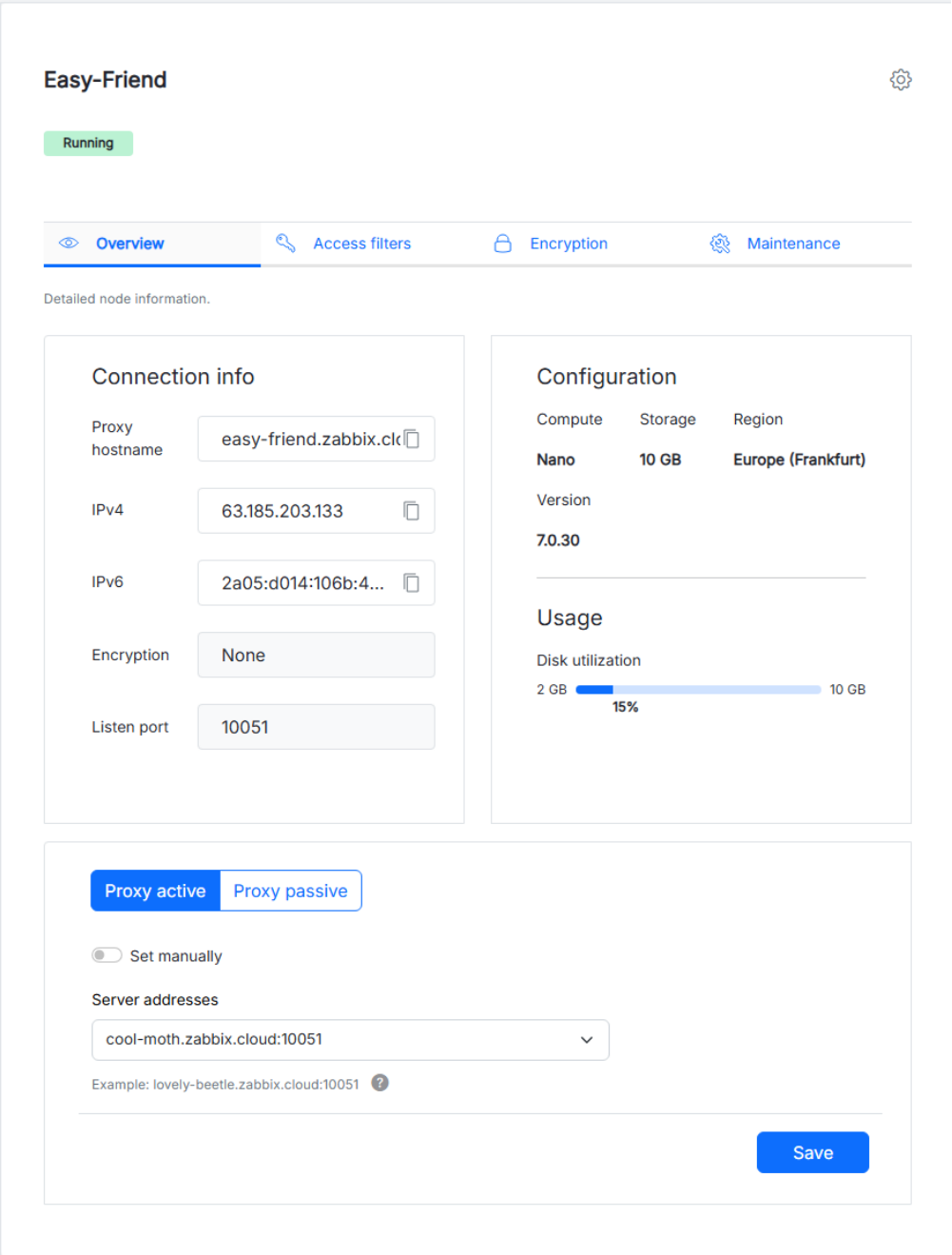
Wykonaj kroki 2–4 dla adresu IP agenta Zabbix używającego aktywnych kontroli.

Węzeł serwera

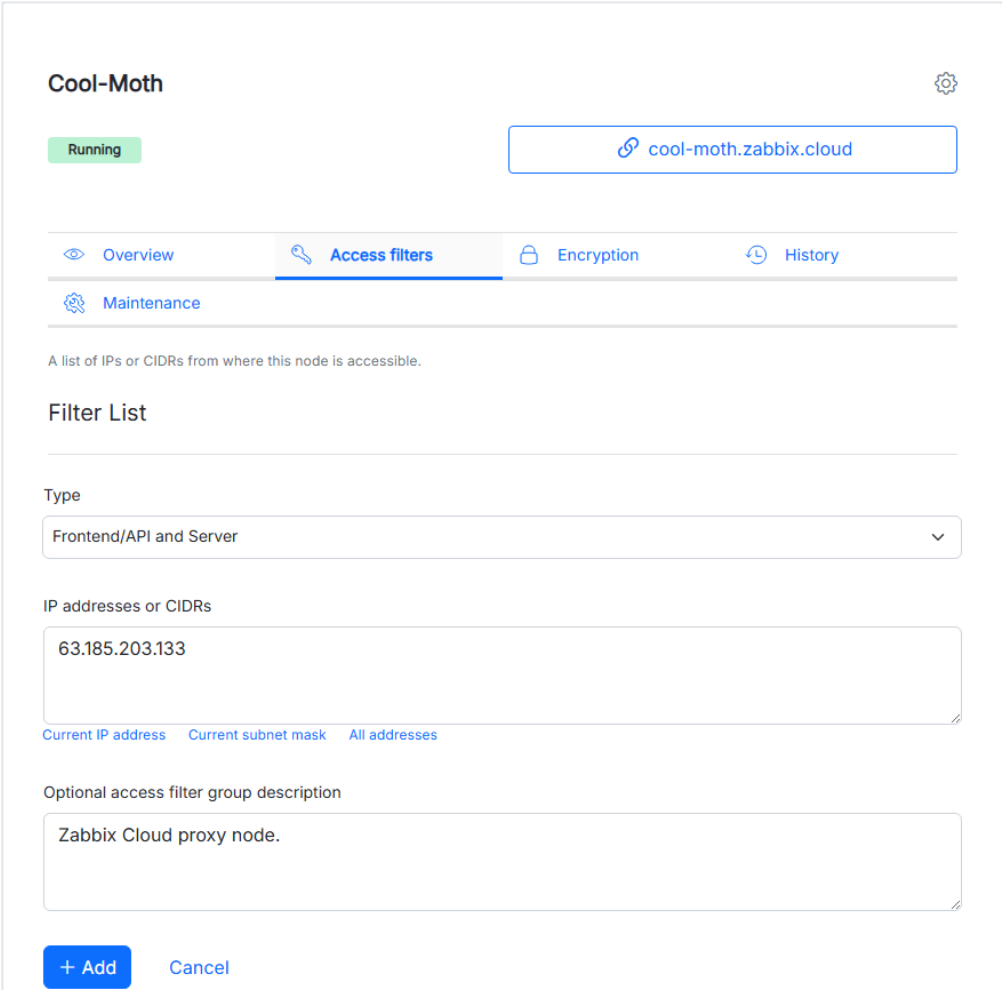
Jeśli serwer Zabbix jest hostowany w Zabbix Cloud, skonfiguruj filtry dostępu również na węźle serwera.

W przypadku proxy działającego w trybie aktywnym dodaj adres IP lub CIDR węzła proxy do filtrów dostępu węzła serwera, ponieważ proxy inicjuje połączenie z serwerem.

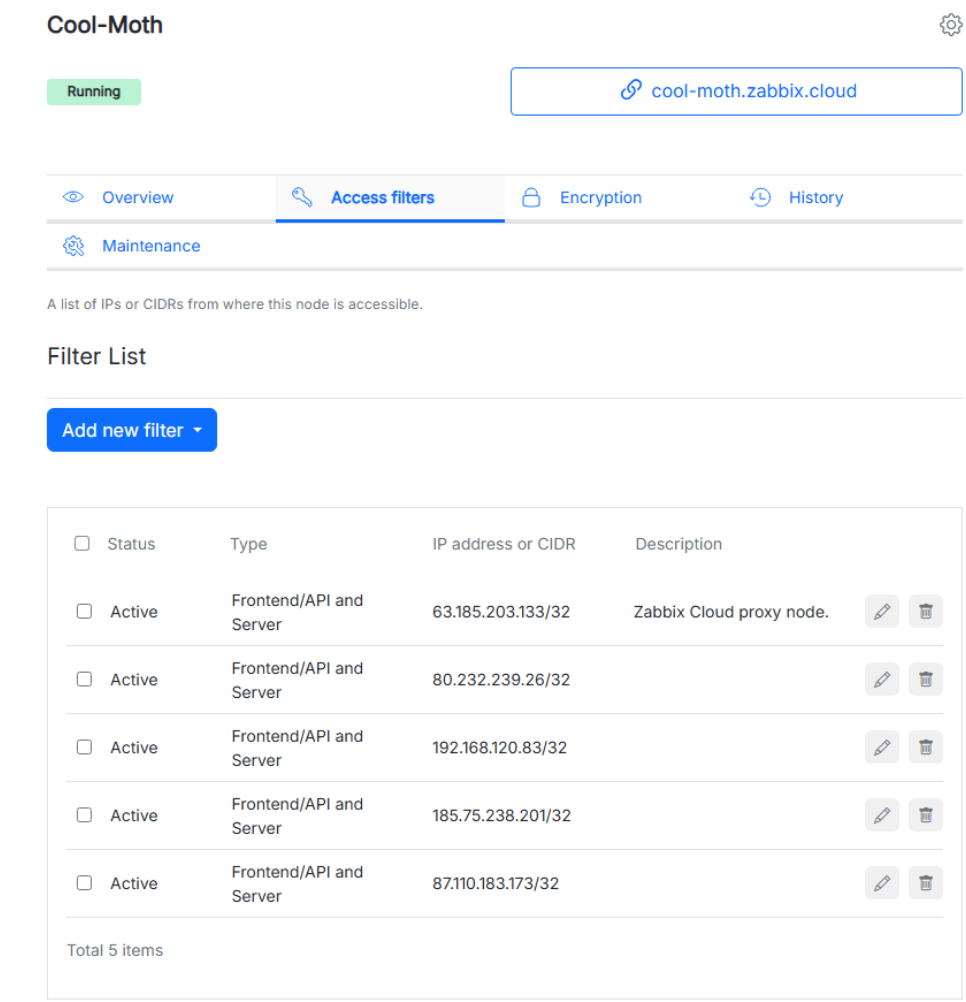
1. Przejdź do karty *Overview* węzła proxy i skopiuj adres IPv4 lub IPv6 węzła.



2. Przejdź do karty *Access filters* węzła serwera, kliknij *Add new filter* > *Add*.
3. Wklej skopiowany adres w polu *IP addresses or CIDRs*, a następnie dodaj opis w polu *Optional access filter group description*.
4. Wybierz odpowiedni typ filtra.



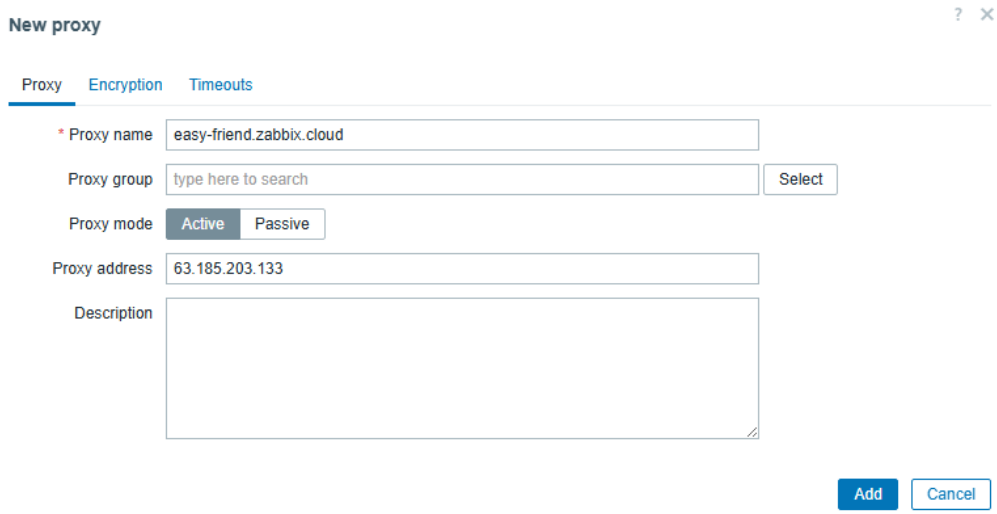
5. Kliknij *Add*.



Konfigurowanie proxy Proxy musi zostać skonfigurowane we frontendzie Zabbix, aby komunikować się z serwerem Zabbix.

Otwórz *Administracja* > *Proxy* we frontendzie Zabbix, kliknij *Utwórz proxy* i skonfiguruj proxy zgodnie z wymaganiami:

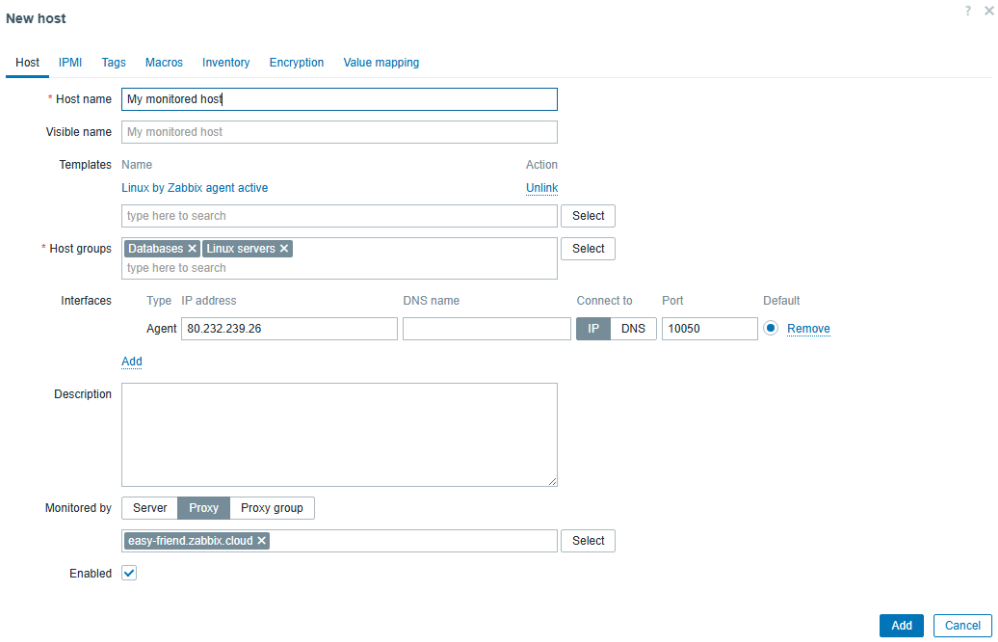
1. Dodaj nazwę proxy w polu *Nazwa proxy*. Musi ona odpowiadać nazwie hosta proxy skonfigurowanej dla proxy. Wpisz ją lub skopiuj z karty *Przegląd* swojego proxy.
2. Wybierz tryb proxy odpowiadający trybowi wybranemu podczas tworzenia proxy.
3. W przypadku aktywnego proxy wprowadź adres IP lub nazwę DNS proxy.
4. Kliknij *Dodaj*.



Więcej informacji znajdziesz w sekcji Monitorowanie rozproszone z użyciem proxy.

Konfigurowanie monitorowanych hostów Dodaj hosty, które powinny być monitorowane za pośrednictwem proxy.

1. Przejdź do *Gromadzenie danych* > *Hosty*.
2. Utwórz host lub otwórz istniejący host.
3. W polu *Monitorowany przez* wybierz *Proxy* i wpisz nazwę proxy, aby wyszukać i wybrać proxy, albo kliknij przycisk *Wybierz*, aby otworzyć listę proxy do wyboru.
4. Skonfiguruj interfejs hosta i inne parametry.
5. Kliknij *Dodaj* lub *Aktualizuj*.



Host skonfigurowany do monitorowania przez proxy wysyła lub odbiera dane monitorowania za pośrednictwem tego proxy, w zależności od typu pozycji i trybu agenta.

Konfigurowanie agent Gdy monitorowany host korzysta z Zabbix agent lub Zabbix agent 2, skonfiguruj agent tak, aby komunikował się z odpowiednim komponentem Zabbix.

Konfiguracja zależy od tego, czy agent używa kontroli *pasywnych*, czy *aktywnych*.

Pasywne sprawdzanie przez agent

W przypadku pasywnych sprawdzeń proxy Zabbix inicjuje połączenia z agentem i żąda danych monitorowania.

Otwórz plik konfiguracyjny agenta Zabbix i zaktualizuj parametr `Server`, aby zezwolić na połączenia z adresu IP proxy Zabbix Cloud.

Na przykład:

```
Server=<proxy_IP>
```

Upewnij się, że monitorowany host zezwala na przychodzące połączenia TCP z proxy na porcie agenta, którym domyślnie jest 10050.

Proxy musi mieć możliwość połączenia się z monitorowanym hostem przez sieć.

Aktywne sprawdzanie przez agent

W przypadku aktywnego sprawdzania agent inicjuje połączenia z proxy Zabbix i wysyła zebrane dane monitorowania.

Otwórz plik konfiguracyjny agenta Zabbix i zaktualizuj parametr `ServerActive`, podając adres proxy Zabbix Cloud.

Na przykład:

`ServerActive=<proxy_IP>:<proxy_listen_port>`

Zastąp `<proxy_IP>` adresem IP proxy Zabbix Cloud, a `<proxy_listen_port>` portem nasłuchiwania skonfigurowanym podczas tworzenia węzła proxy. Domyślny port nasłuchiwania to 10051.

Jeśli w parametrze `ServerActive` określono proxy, agent pobiera konfigurację aktywnego sprawdzania z proxy, a nie bezpośrednio z serwera Zabbix.

Upewnij się, że monitorowany host może nawiązać wychodzące połączenie TCP z proxy na skonfigurowanym porcie nasłuchiwania.

Więcej informacji o pasywnym i aktywnym sprawdzaniu przez agenta znajdziesz w sekcji Pasywne i aktywne sprawdzanie przez agenta.

Zewnętrzny serwer Zabbix Proxy Zabbix Cloud może również komunikować się z serwerem Zabbix wdrożonym poza Zabbix Cloud.

W takim przypadku zewnętrzny serwer Zabbix musi mieć możliwość komunikowania się z proxy w chmurze zgodnie z wybranym trybem proxy.

W przypadku proxy w trybie aktywnym proxy inicjuje połączenie z zewnętrznym serwerem Zabbix. Skonfiguruj zewnętrzny serwer Zabbix tak, aby akceptował połączenia z proxy, i upewnij się, że proxy może połączyć się z serwerem przez sieć.

W przypadku proxy pasywnego serwer Zabbix inicjuje połączenie z proxy. Dodaj adres IP lub podsieć zewnętrznego serwera Zabbix do filtrów dostępu węzła proxy i upewnij się, że serwer może połączyć się z proxy przez sieć.

Zewnętrzny serwer Zabbix nie korzysta z filtrów dostępu Zabbix Cloud. Zamiast tego skonfiguruj wymagane reguły zapory sieciowej i sieci w środowisku zewnętrznym.

Monitorowane hosty mogą znajdować się w tym samym środowisku co zewnętrzny serwer Zabbix, w innej sieci lub w sieci dostępnej dla proxy.

Gdy agenty używają aktywnych kontroli, skonfiguruj ich parametr `ServerActive`, aby wskazywał proxy Zabbix Cloud. Gdy agenty używają pasywnych kontroli, skonfiguruj ich parametr `Server`, aby zezwalał na połączenia z proxy.

Weryfikacja konfiguracji Po skonfigurowaniu proxy, agentów, filtrów dostępu i łączności sieciowej sprawdź, czy proxy komunikuje się z serwerem Zabbix.

1. Przejdź do *Administracja* > *Proxy*.
2. Sprawdź *Stan* proxy.
3. Przejdź do *Monitorowanie* > *Najnowsze dane* i sprawdź, czy są dostępne najnowsze wartości pozycji.

Jeśli proxy nie komunikuje się z serwerem, sprawdź:

- tryb proxy;
- adres i port serwera;
- filtry dostępu proxy;
- filtry dostępu serwera, gdy serwer jest hostowany w Zabbix Cloud;
- konfigurację agenta;
- reguły zapory sieciowej i routingu;
- adres skonfigurowany w `Server` lub `ServerActive`.

Aplikacja mobilna

Aplikacja mobilna

Aplikacja mobilna Zabbix umożliwia monitorowanie środowiska Zabbix z urządzenia mobilnego. Wyświetlaj problemy, sprawdzaj ich szczegóły, aktualizuj problemy i otrzymuj powiadomienia push, gdy coś wymaga uwagi.

Pobierz aplikację (wkrótce)

Skonfiguruj Zabbix dla aplikacji

Skonfiguruj adapter Zabbix bridge Zainstaluj adapter Zabbix bridge, aby umożliwić urządzeniom mobilnym łączenie się z Zabbix.

Dodawanie urządzeń Dodaj urządzenie mobilne w frontendzie Zabbix, aby połączyć je z serwerem Zabbix.

Konfigurowanie powiadomień push Otrzymuj alerty na urządzeniu mobilnym, gdy wystąpi problem.

Dowiedz się więcej o aplikacji

Przegląd aplikacji Poznaj podstawy korzystania z aplikacji mobilnej Zabbix.

Bezpieczny dostęp mobilny Zabezpiecz aplikację, urządzenia mobilne i adapter Zabbix bridge albo unieważnij dostęp z zagubionych lub przejętych urządzeń.

Konfigurowanie Zabbix na potrzeby aplikacji

Aby korzystać z aplikacji mobilnej Zabbix (wkrótce dostępnej), musisz najpierw:

1. Skonfigurować adapter Zabbix bridge.
2. Skonfigurować powiadomienia push.
3. Dodać urządzenie mobilne w frontendzie Zabbix.

Poniższe sekcje opisują każdy z tych kroków.

Po zakończeniu konfiguracji zapoznaj się z sekcją Bezpieczny dostęp mobilny, aby zapewnić bezpieczeństwo aplikacji i adaptera Zabbix bridge.

Konfiguracja adaptera Zabbix bridge Aby umożliwić komunikację serwera Zabbix z aplikacją mobilną, należy skonfigurować adapter Zabbix bridge.

Adapter bridge można zainstalować w tym samym systemie co serwer Zabbix lub w innym systemie, korzystając z pakietów:

> **Attention:**
> Poniższe kroki dotyczą systemu Ubuntu 24.04. W przypadku innych systemów operacyjnych pobierz polecenia instalacji repozytorium i pakietów (np. `wget`, `dpkg`, `apt`) ze strony Get Zabbix dla używanego systemu operacyjnego i użyj ich do zainstalowania pakietu `zabbix-bridge-adapter`.

1. Rozpocznij nową sesję powłoki z uprawnieniami root:

```
sudo -s
```

2. Zainstaluj repozytorium 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. Zainstaluj adapter Zabbix bridge:

```
apt install zabbix-bridge-adapter
```

4. Edytuj plik konfiguracyjny serwera Zabbix:

```
EnableMobileDevices=1
BridgeAdapterURL=https://bridge-adapter.example.com:10054/rpc
```

5. Uruchom ponownie serwer Zabbix, na przykład (zobacz inne opcje):

```
systemctl restart zabbix-server
```

6. Skopiuj identyfikator serwera Zabbix z sekcji *Raporty* > *Informacje o systemie*; będzie on potrzebny w następnym kroku.

7. Edytuj plik konfiguracyjny adaptera Zabbix bridge:

```
ZabbixAPIURL=https://example.com/zabbix/api_jsonrpc.php
ServerId=019fb1f9-22bd-74ea-bc03-51efb8b76aba
DataDir=/var/lib/zabbix-bridge-adapter
```

> **Attention:**
> Katalog określony w `DataDir` musi mieć uprawnienia 0700, a wszystkie przechowywane w nim pliki muszą mieć uprawnienia 0600.

8. Zapoznaj się z umową licencyjną użytkownika końcowego (EULA) adaptera Zabbix bridge i zaakceptuj ją:

```
zabbix_bridge_adapter license show
sudo -u zabbix-bridge-adapter zabbix_bridge_adapter -c /etc/zabbix/zabbix_bridge_adapter.conf license acce
```

9. Uruchom adapter Zabbix bridge:

```
systemctl start zabbix-bridge-adapter
```

Attention:
Po uruchomieniu adapter Zabbix bridge używa identyfikatora serwera do zarejestrowania serwera Zabbix w centralnie hostowanej usłudze Zabbix Bridge. Stan rejestracji jest przechowywany w pliku stanu w katalogu `DataDir`. Jeśli adapter bridge zostanie ponownie zainstalowany, a plik stanu zostanie utracony, ponowna rejestracja tego samego serwera Zabbix może się nie powieść. Aby tego uniknąć, przed ponowną instalacją adaptera bridge utwórz kopię zapasową pliku stanu, a następnie przywróć go po instalacji.

Konfigurowanie powiadomień push Aby otrzymywać alerty na urządzeniu mobilnym, gdy coś wymaga uwagi w monitorowanym środowisku:

1. Utwórz typ nośnika powiadomień push (lub włącz domyślny).
2. Utwórz akcję (lub włącz domyślną akcję wyzwalacza *Zgłaszaj problemy administratorom Zabbix*).
3. Dodaj urządzenie, aby automatycznie dodać typ nośnika do swojego użytkownika (lub ręcznie dodaj nośnik użytkownika).

Dodawanie urządzeń Aby korzystać z aplikacji mobilnej Zabbix na swoim urządzeniu, musisz dodać to urządzenie w frontend Zabbix. Instrukcje znajdziesz w:

- *Użytkownicy* > *Urządzenia* (jako *Super admin* dodaj własne lub mobilne urządzenia innych użytkowników)
- *Ustawienia użytkownika* > *Urządzenia* (jako dowolny użytkownik dodaj własne urządzenia mobilne)

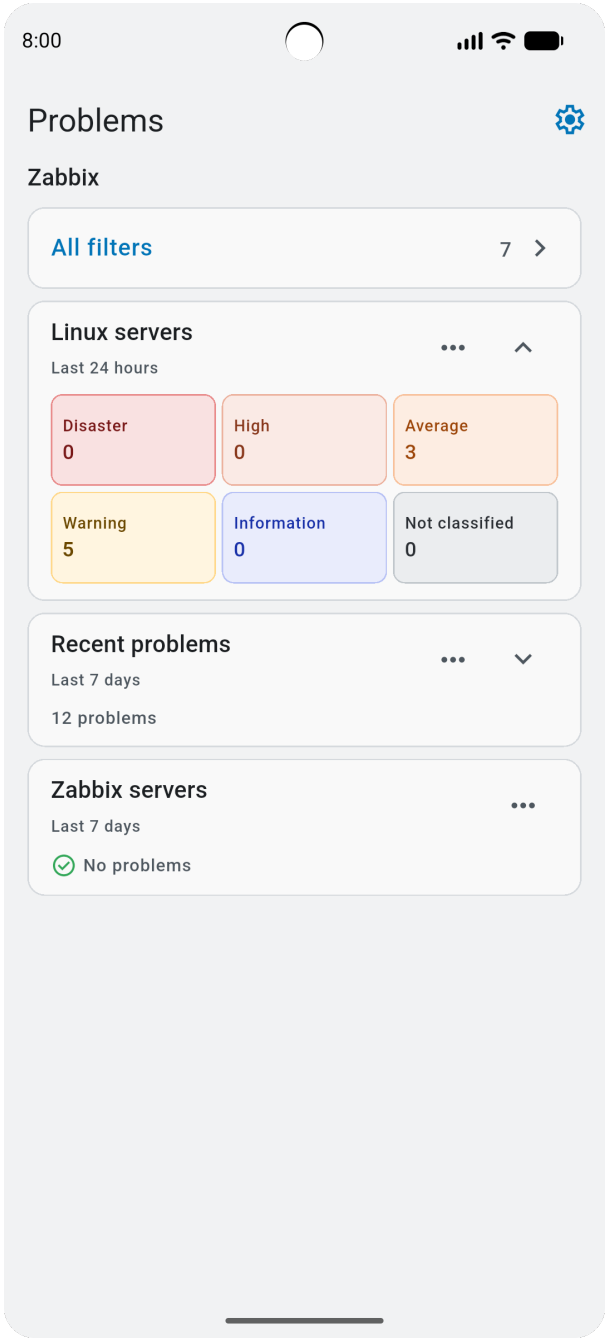
Przegląd aplikacji

Aplikacja mobilna Zabbix (już wkrótce) umożliwia monitorowanie środowiska Zabbix z urządzenia mobilnego. Wyświetlaj problemy, sprawdzaj ich szczegóły, aktualizuj problemy i otrzymuj powiadomienia push, gdy coś wymaga uwagi.

Ta strona zawiera przegląd aplikacji mobilnej oraz informacje o sposobie jej używania.

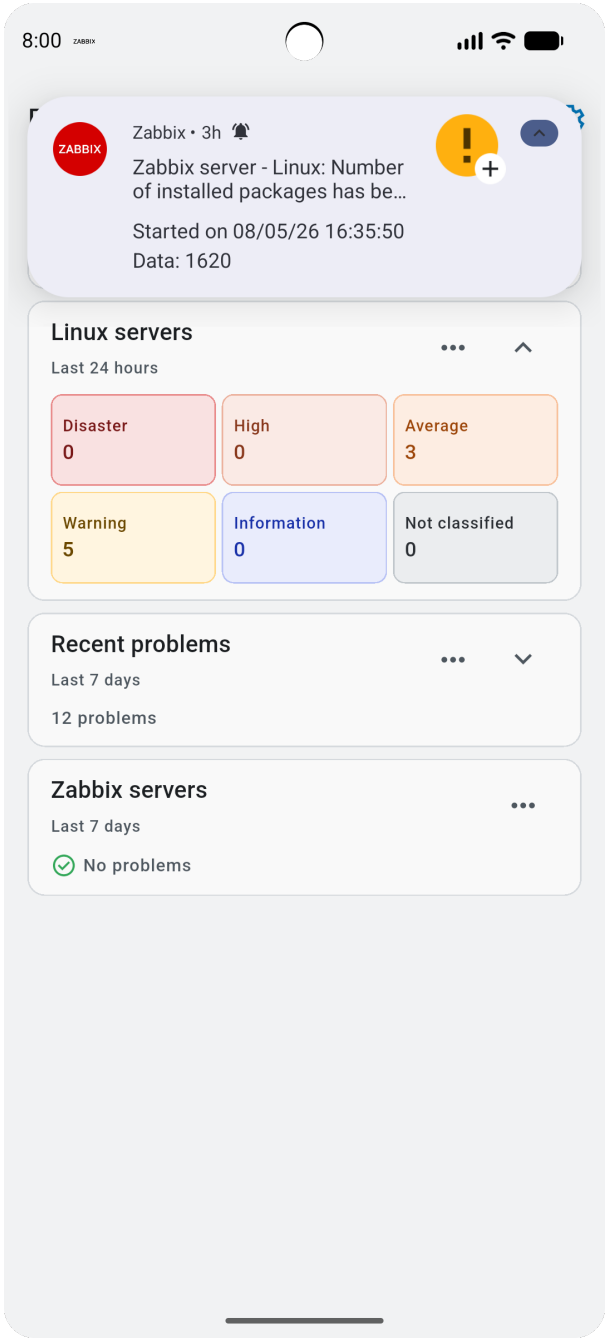
Problemy Ekran *Problemy* przedstawia przegląd problemów na podstawie przypiętych filtrów. Zawiera:

- Ikonę *Ustawienia* - dotknij jej, aby wyświetlić ustawienia aplikacji.
- *Wszystkie filtry* - dotknij, aby wyświetlić filtry i zarządzać nimi.
- Nazwę serwera Zabbix (lub *Zabbix*, jeśli nie skonfigurowano jej podczas instalacji interfejsu webowego Zabbix).
- Przypięte filtry - dotknij ich nazwy lub poziomu ważności, aby otworzyć ekran *Lista problemów*. Dotknij menu z trzema kropkami, aby zarządzać filtrem (edytować, zduplikować, odpiąć lub usunąć), albo dotknij strzałki, aby go rozwinąć lub zwinąć. Pamiętaj, że przypięte filtry są uporządkowane alfabetycznie, a filtru *Najnowsze problemy* nie można edytować ani usunąć.



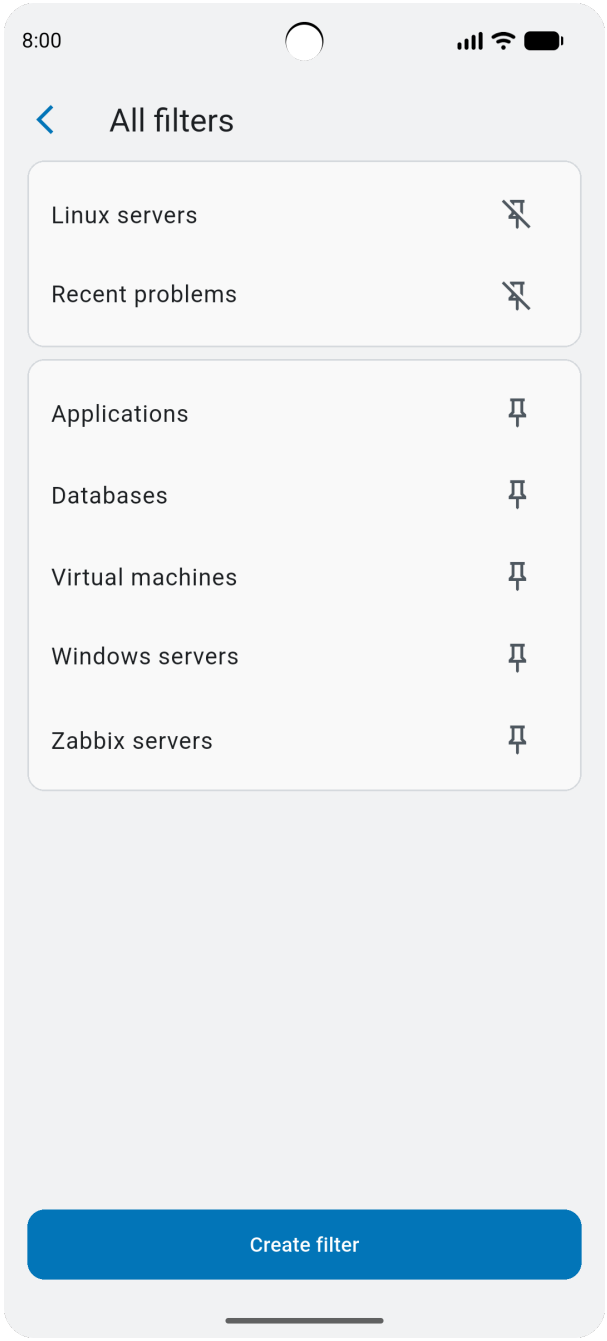
Powiadomienia Gdy serwer Zabbix wykryje problem lub gdy problem zostanie zaktualizowany albo rozwiązany, może wysłać powiadomienie na urządzenie mobilne. Powiadomienie zawiera host problemu, jego nazwę oraz komunikat wygenerowany na podstawie szablonów wiadomości skonfigurowanych dla typu nośnika powiadomienie push.

Dotknij powiadomienia, aby otworzyć ekran *Szczegóły problemu*.



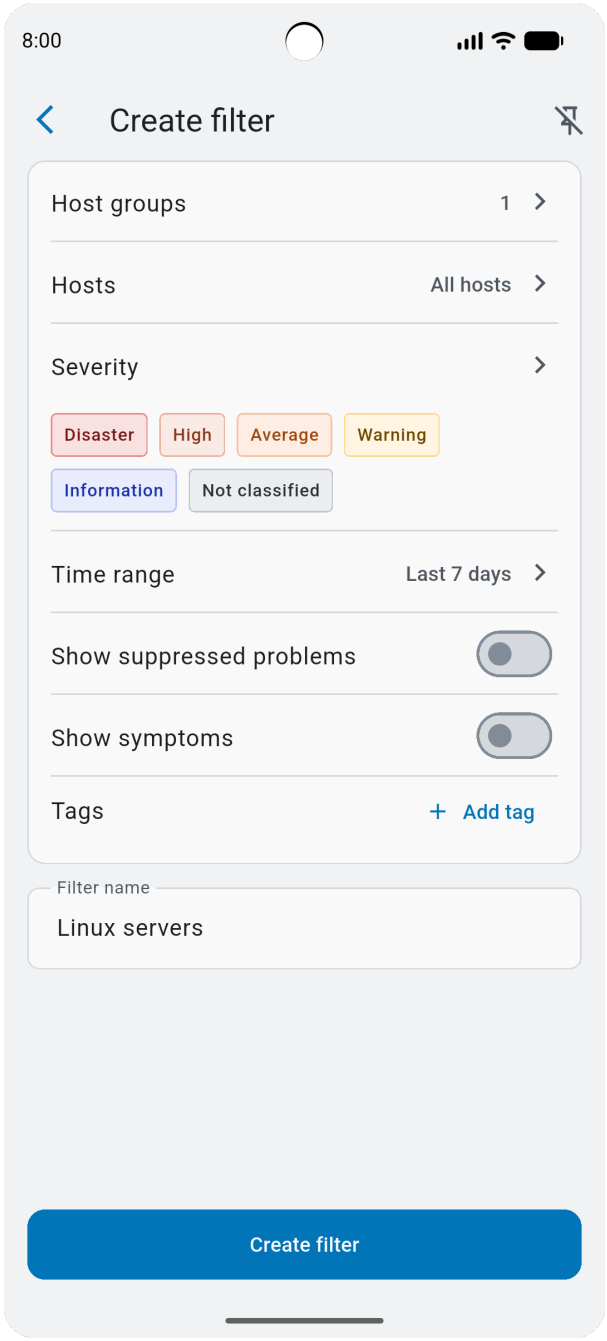
Wszystkie filtry Ekran *Wszystkie filtry* umożliwia zarządzanie filtrami. Zawiera:

- Nazwy filtrów - dotknij, aby edytować filtr.
- Ikony pinezek - dotknij, aby przypiąć filtr do ekranu *Problemy* lub go odpiąć.
- *Utwórz filtr* - dotknij, aby utworzyć nowy filtr.



Tworzenie filtra Ekran *Tworzenie filtra* umożliwia utworzenie filtra z opcjami podobnymi do dostępnych w sekcji *Monitoring* > *Problemy* (zobacz szczegóły). Możesz także ustawić *Zakres czasu*, aby określić, jak długo mają być wyświetlane nierozwiązane problemy.

Dotknij *Utwórz filtr*, aby zapisać filtr. Nowe filtry są przypinane do ekranu *Problemy*; dotknij ikony pinezki, aby to zmienić.



Lista problemów Ekran *Lista problemów* wyświetla problemy pasujące do filtra (np. *Ostatnie problemy*).

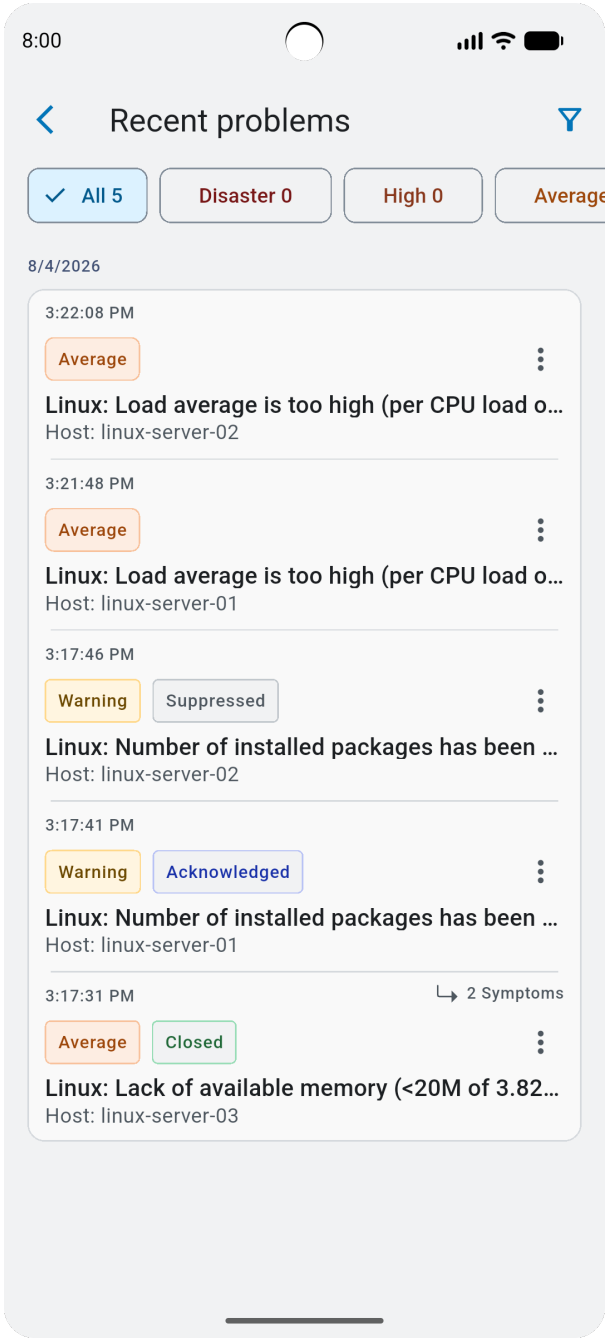
Dotknij ikony filtra lub poziomu ważności problemu, aby dodatkowo zawęzić listę.

Dla każdego problemu możesz zobaczyć:

- Godzinę i datę rozpoczęcia.
- Poziom ważności, nazwę i host.
- Czy został wyciszony (wyświetlane, jeśli włączono opcję *Pokaż wyciszone problemy*).
- Czy został potwierdzony lub niedawno zamknięty (automatycznie rozwiązany lub zamknięty ręcznie).
- Czy ma objawy - dotknij, aby wyświetlić listę problemów będących objawami.

Dotknij problemu, aby otworzyć ekran *Szczegóły problemu*.

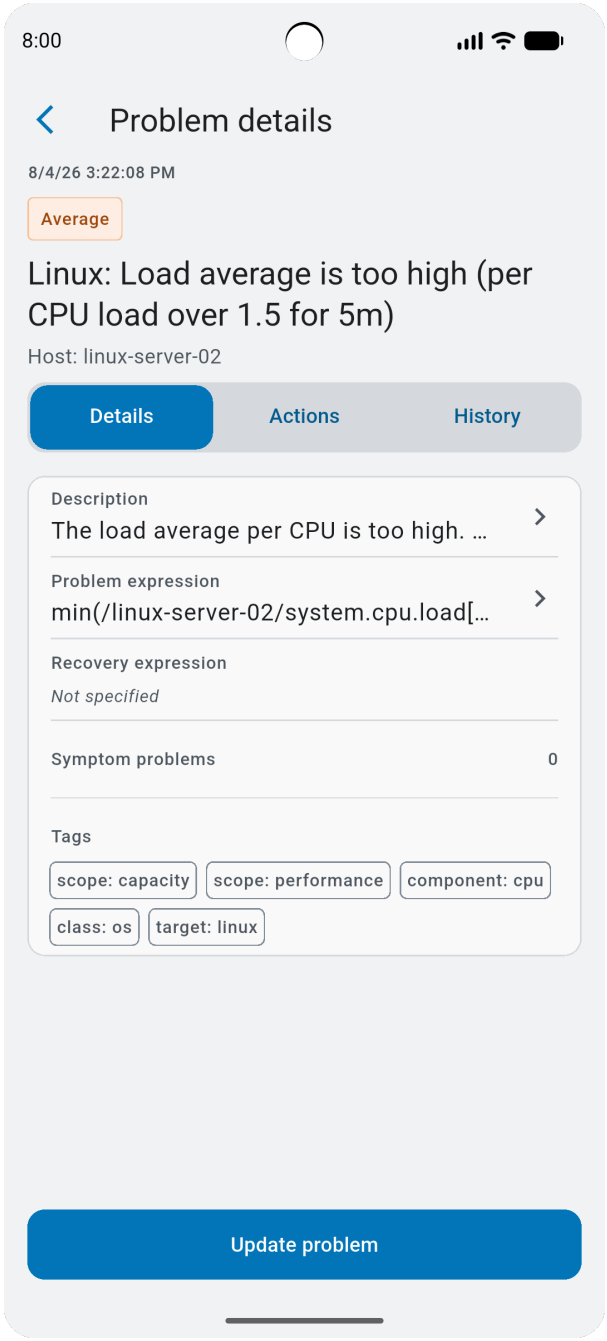
Dotknij menu z trzema kropkami, aby szybko potwierdzić problem lub wyciszyć go na wybrany czas.



Szczegóły problemu Ekran *Szczegóły problemu* wyświetla szczegóły problemu, działania i historię.

- Karta *Szczegóły* - wyświetla opis problemu, wyrażenia, problemy objawowe (dotknij, aby wyświetlić listę problemów) oraz znaczniki.
- Karta *Działania* - wyświetla oś czasu problemu: kiedy się rozpoczął lub został potwierdzony, czy wysłano powiadomienie oraz inne działania.
- Karta *Historia* - wyświetla wiek problemu, czas trwania i czas przywrócenia (lub *Niedostępne*, jeśli problem nadal trwa).

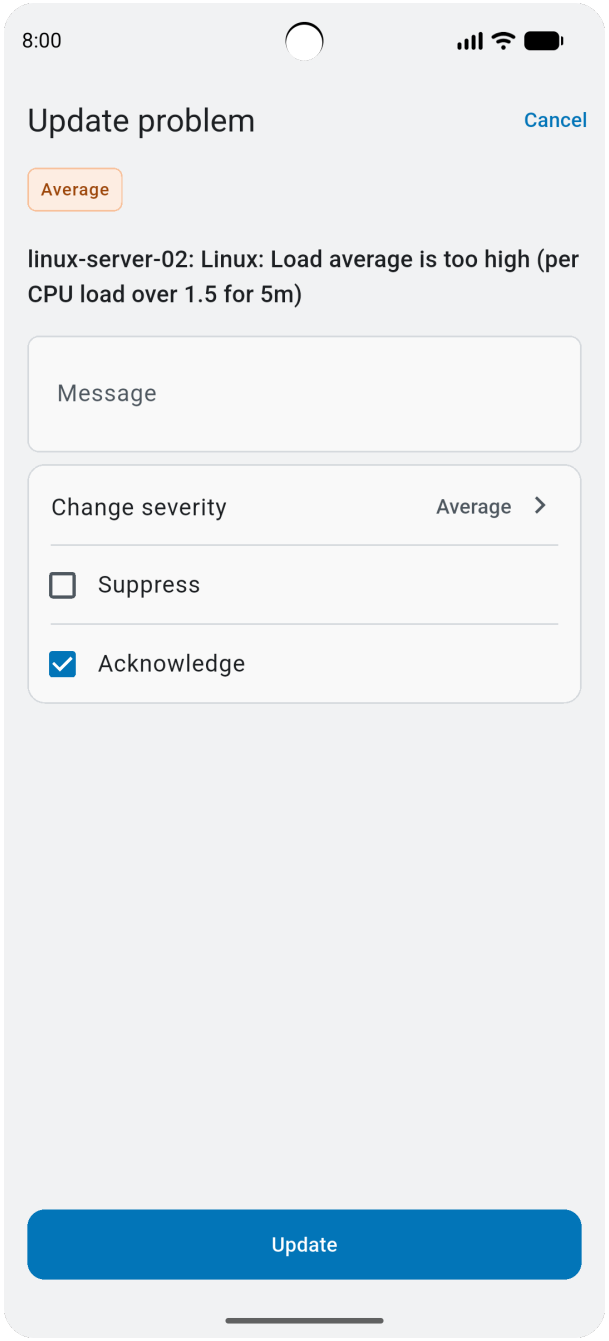
Dotknij opcji *Aktualizuj problem*, aby zaktualizować problem.



Aktualizuj problem Ekran *Aktualizuj problem* umożliwia aktualizację problemu:

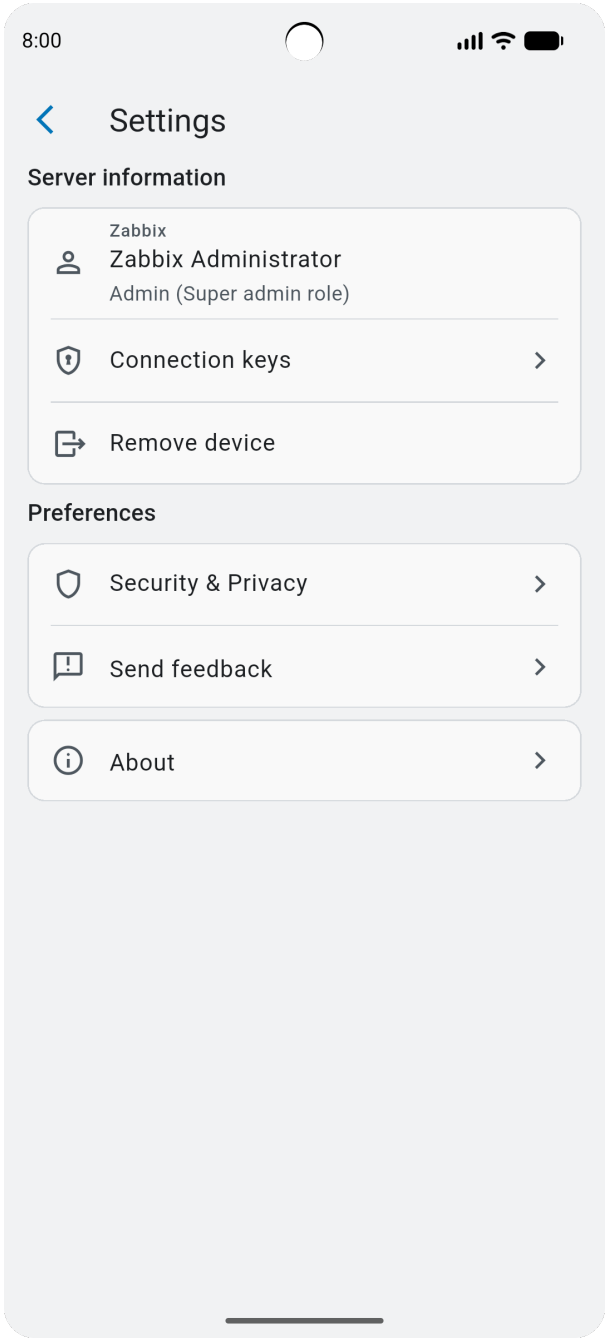
- Dodanie wiadomości.
- Zmianę ważności problemu.
- Wyciszenie problemu na wybrany czas.
- Potwierdzenie lub anulowanie potwierdzenia problemu.
- Zamknięcie problemu.

Dotknij *Aktualizuj*, aby zapisać zmiany.



Ustawienia Ekran *Ustawienia* umożliwia dostęp do ustawień aplikacji. Zawiera:

- Nazwę serwera Zabbix (lub *Zabbix*, jeśli nie została skonfigurowana podczas instalacji interfejsu webowego Zabbix).
- Nazwę, typ i rolę użytkownika, do którego dodano urządzenie mobilne.
- *Klucze połączenia* - otwórz, aby wyświetlić i skopiować identyfikator serwera, identyfikator urządzenia oraz inne szczegóły.
- *Usuń urządzenie* - dotknij, aby usunąć urządzenie mobilne z Zabbix.
- *Bezpieczeństwo i prywatność* - otwórz, aby włączyć blokadę aplikacji i zarządzać udostępnianiem danych diagnostycznych.
- *Wyślij opinię* - otwórz, aby wysłać opinię do Zabbix.
- *Informacje* - otwórz, aby wyświetlić wersję aplikacji i inne informacje o aplikacji.



Bezpieczny dostęp mobilny

Ta strona zawiera najlepsze praktyki dotyczące zapewnienia bezpieczeństwa aplikacji mobilnej Zabbix i adaptera Zabbix bridge.

Ochrona urządzeń mobilnych Aby uniemożliwić innym osobom otwieranie aplikacji na urządzeniu mobilnym, włącz blokadę aplikacji w sekcji *Ustawienia*.

Jeśli urządzenie zostało zgubione lub przejęte, usuń je w frontendzie Zabbix. Spowoduje to unieważnienie i usunięcie danych uwierzytelniających dostępu, dzięki czemu urządzenie nie będzie już mogło uzyskać dostępu do Zabbix. W zależności od uprawnień przypisanych do roli użytkownika urządzenia można usuwać w następujących sekcjach:

- *Ustawienia użytkownika > Urządzenia* - każdy użytkownik może usuwać własne urządzenia (wymaga uprawnienia *Zarządzanie własnymi urządzeniami*).
- *Użytkownicy > Urządzenia* - *Super admin* może usuwać urządzenia dowolnego użytkownika (wymaga uprawnienia *Zarządzanie urządzeniami użytkowników*).

Można również ograniczyć zakres Zabbix, do którego można uzyskać dostęp z urządzenia mobilnego. W tym celu przypisz użytkownikom rolę użytkownika, która przyznaje tylko uprawnienia wymagane przez aplikację. Minimalne uprawnienia są następujące:

Sekcja	Konfiguracja
Dostęp do elementów interfejsu użytkownika	Wyczyść wszystkie pola wyboru.
Dostęp do usług	Ustaw opcje *Dostęp do usług w trybie odczytu i zapisu* oraz *Dostęp do usług w trybie tylko do odczytu* na `None`.
Dostęp do modułów	Wyczyść wszystkie pola wyboru.
Dostęp do API	Wyczyść pole wyboru *Włączone*.
Dostęp do działań	Wyczyść wszystkie pola wyboru. Aby umożliwić użytkownikom aktualizowanie problemów w aplikacji, zaznacz również wymagane działania: - *Dodawanie komentarzy do problemów* - *Zmiana ważności* - *Potwierdzanie problemów* - *Wstrzymywanie problemów* - *Zamykanie problemów* - *Tworzenie i edytowanie własnych mediów* Pamiętaj, że problemy widoczne dla użytkowników w aplikacji zależą również od ich dostępu do hostów.
Urządzenia	Zaznacz pole wyboru *Włączone*. Aby umożliwić użytkownikom dodawanie i usuwanie własnych urządzeń mobilnych, zaznacz również pole wyboru *Zarządzanie własnymi urządzeniami*; w przeciwnym razie *Super admin* musi zarządzać ich urządzeniami w sekcji *Użytkownicy > Urządzenia*.

Ochrona adaptera Zabbix bridge Adapter Zabbix bridge przechowuje swoje dane (plik stanu i klucze szyfrowania) w katalogu określonym w parametrze `DataDir` pliku konfiguracyjnego adaptera Zabbix bridge. Upewnij się, że katalog ma uprawnienia 0700, a wszystkie przechowywane w nim pliki mają uprawnienia 0600.

Adapter Zabbix bridge przekazuje żądania z urządzeń mobilnych do API Zabbix. Aby ograniczyć metody API, które można wywoływać za pośrednictwem adaptera Zabbix bridge, użyj jego parametru `AllowedAPIMethods`. Na przykład zezwól tylko na minimalny zestaw metod wymaganych przez aplikację mobilną lub na jeszcze mniejszy zestaw, jeśli nie chcesz, aby niektóre funkcje aplikacji mobilnej były używane (np. nie zezwalaj na `event.acknowledge`, jeśli użytkownicy aplikacji mobilnej nie powinni aktualizować problemów).

Centrum dewelopera

Centrum deweloperskie

Rozpocznij tworzenie niestandardowych rozszerzeń dla Zabbix, korzystając ze wskazówek dotyczących wtyczek, modułów i widgetów.

Twórz na Zabbix

Wtyczki Przegląd sposobu tworzenia i zarządzania wtyczkami Zabbix w celu rozszerzenia funkcjonalności lub integracji z systemami zewnętrznymi.

Biblioteka Python dla Zabbix Zasoby dotyczące korzystania z oficjalnej biblioteki `zabbix_utils` do interakcji z API Zabbix i tworzenia integracji.

Moduły frontend Przewodniki i materiały referencyjne dotyczące tworzenia niestandardowych modułów, które rozszerzają lub modyfikują frontend Zabbix, aby dostosować go do konkretnych przypadków użycia.

Widgety Omówienie struktury i logiki widgetów wraz z instrukcjami tworzenia niestandardowych elementów pulpitu dostosowanych do Twoich potrzeb.

Tworzenie wtyczki

Ta strona jest przeznaczona dla osób zainteresowanych opracowaniem własnej wtyczki (niezależnego modułu zbierającego dane) dla Zabbix agent 2.

Niestandardowa wtyczka rozszerza funkcjonalność Zabbix agent 2 poza dostarczane wbudowane wtyczki i wtyczki ładowalne.

Każda wtyczka jest pakietem *Go*, który definiuje strukturę i implementuje jeden lub kilka interfejsów (*Exporter*, *Configurator*, *Runner*). Więcej informacji można znaleźć w sekcjach Interfejsy wtyczek i Diagram połączeń.

Ten przewodnik pomoże Ci utworzyć niestandardową wtyczkę ładowalną.

Dodatkowe wskazówki można znaleźć również w repozytoriach:

- Przykładowa wtyczka
- Istniejące wtyczki ładowalne

Co utworzysz To jest samouczek krok po kroku dotyczący tworzenia prostego ładowalnego pluginu o nazwie **MyIP**. Ten plugin zaimplementuje pojedynczy klucz pozycji (`myip`), zwracający zewnętrzny adres IP hosta agenta Zabbix.

Krok 1: Konfiguracja

1. Wtyczka jest standardowym modułem Go. Zacznij od zainicjowania pliku `go.mod` w katalogu wtyczki, aby śledzić zależności wtyczki:

```
cd path/to/plugins/myip # Przejdź do katalogu swojej wtyczki
go mod init myip
```

2. Zainstaluj obowiązkową zależność Zabbix Go SDK (`golang.zabbix.com/sdk`):

```
go get golang.zabbix.com/sdk@$LATEST_COMMIT_HASH
```

Zastąp `$LATEST_COMMIT_HASH` najnowszym hashem commita `HEAD` z repozytorium `golang.zabbix.com/sdk` w odpowiedniej gałęzi wydania. Na przykład:

```
go get golang.zabbix.com/sdk@af85407
```

Pamiętaj, że wersjonowanie `golang.zabbix.com/sdk` nie jest obecnie obsługiwane, ale może się to zmienić w przyszłości.

Dodatkowe zależności można instalować w razie potrzeby za pomocą `go get`.

3. Utwórz pusty plik `main.go` dla kodu źródłowego wtyczki:

```
touch main.go
```

Początkowa konfiguracja jest teraz zakończona i wtyczka jest gotowa do dalszego tworzenia.

Krok 2: Struktura wtyczki Moduł `golang.zabbix.com/sdk`, zainstalowany w poprzednim kroku, udostępnia niezbędne środowisko do tworzenia wtyczek i zapewnia spójną strukturę wszystkich wtyczek.

1. Skonfiguruj podstawowy przepływ wykonywania.

Zacznij od zdefiniowania głównego przepływu wykonywania wtyczki. Dodaj następujący kod do pliku `main.go`:

```
package main

func main() {
    err := run()
    if err != nil {
        panic(err)
    }
}

func run() error {
    return nil
}
```

Ustanawia to podstawowy przepływ wykonywania wtyczki. Funkcja run będzie później zawierać główną logikę wtyczki.

2. Zapoznaj się z interfejsami wtyczki.

Wtyczka Zabbix agent 2 powinna być reprezentowana przez strukturę implementującą interfejsy z pakietu `golang.zabbix.com/sdk/plugi`

- *Accessor* - definiuje podstawowe metody, które muszą implementować wszystkie wtyczki, takie jak ustawianie nazwy wtyczki i obsługa limitów czasu kluczy pozycji.

- Jeden lub więcej z następujących funkcjonalnych interfejsów wtyczki:
  - *Exporter* - wykonuje odpytywanie i zwraca wartość (lub wartości), nic albo błąd; często jest używany razem z interfejsem *Collector*.
  - *Collector* - zarządza okresowym zbieraniem danych.
  - *Runner* - definiuje procedury uruchamiania i zamykania wtyczki.
  - *Watcher* - umożliwia niezależne odpytywanie metryk z pominięciem wewnętrznego harmonogramu agenta; jest przydatny w monitorowaniu opartym na pułapkach lub zdarzeniach.
  - *Configurator* - definiuje sposób odczytywania i stosowania ustawień konfiguracji wtyczki.

Możesz zaimplementować te interfejsy samodzielnie albo użyć domyślnych implementacji udostępnianych przez Zabbix Go SDK i zmodyfikować je zgodnie z potrzebami. W tym samouczku używane są implementacje domyślne.

3. Utwórz strukturę wtyczki.

Teraz zaimportuj pakiet *plugin* i utwórz strukturę `myIP`, która osadza strukturę `plugin.Base`:

```
import "golang.zabbix.com/sdk/plugin"

type myIP struct {
    plugin.Base
}
```

Struktura myIP obecnie spełnia interfejs Accessor. Metoda implementująca jeden z funkcjonalnych interfejsów wtyczki, *Exporter*, zostanie dodana w dalszej części samouczka.

Krok 3: Zdefiniuj klucze pozycji Twoja wtyczka potrzebuje kluczy pozycji do zbierania danych i przekazywania ich do serwera Zabbix lub proxy.

1. Zaimportuj pakiet *errs* do obsługi błędów:

```
import "golang.zabbix.com/sdk/errs"
```

2. Zarejestruj klucze pozycji za pomocą funkcji `plugin.RegisterMetrics()` wewnątrz funkcji `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
}
```

Aby zarejestrować kilka kluczy pozycji, powtórz parametry *nazwa metryki* i *opis* dla każdej metryki.
Na przykład:

```
plugin.RegisterMetrics(&impl, "Myip", "metric.one", "Metric one description.", "metric.two", "Metric two d
```

Krok 4: Skonfiguruj handler Handler ułatwia komunikację między agent a wtyczką.

1. Zaimportuj pakiet *container*:

```
import "golang.zabbix.com/sdk/plugin/container"
```

2. W funkcji `run()` dodaj kod tworzący i konfigurujący 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
}
```

Krok 5: Implementacja zbierania danych Zbieranie danych odbywa się za pośrednictwem interfejsu Exporter, który opisuje metodę `Export`:

```
func Export(
  key string,             // Klucz pozycji do zebrania.
  params []string,        // Argumenty przekazane do klucza pozycji (`myip[arg1, arg2]`).
  context ContextProvider // Metadane dotyczące zbierania danych klucza pozycji.
) (any, error)
```

1. Zaimportuj wymagane pakiety do żądań HTTP i odczytu odpowiedzi:

```
import (
    "io"
    "net/http"
)
```

2. Zaimplementuj metodę `Export` dla struktury `myIP`:

```
func (p *myIP) Export(
    key string, params []string, context plugin.ContextProvider,
) (any, error) {
    // Plugin może używać różnej logiki zbierania danych w zależności od parametru `key`.
    // Ta implementacja jedynie sprawdza, czy podany `key` jest obsługiwany.
    if key != "myip" {
        return nil, errs.Errorf("unknown item key %q", key)
    }

    // Log zostanie przekazany do logu agent 2.
    p.Infof(
        "received request to handle %q key with %d parameters",
        key,
        len(params),
    )

    // Zbierz dane i zwróć je.
```

```
    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
}
```

Krok 6: Zbuduj i skonfiguruj plugin

1. Aby zbudować plugin, uruchom:

```
go mod tidy
go build
```

Powinno to utworzyć plik wykonywalny `myip` w bieżącym katalogu.

2. Skonfiguruj agent 2 Zabbixa, aby używał pluginu:

```
echo "Plugins.MyIP.System.Path=$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE" > /etc/zabbix/zabbix_agent2.d/plugins.
```

Zastąp `$PATH_TO_THE_MYIP_PLUGIN_EXECUTABLE` ścieżką do `myip` utworzonego w kroku 5.

Nazwa pluginu w nazwie parametru konfiguracji (*MyIP* w tym samouczku) musi być zgodna z nazwą pluginu zdefiniowaną w funkcji *plugin.RegisterMetrics()*.

3. Aby przetestować plugin i jego pozycję `myip`, uruchom:

```
zabbix_agent2 -c /etc/zabbix/zabbix_agent2.conf -t myip
```

Wynik powinien zawierać zewnętrzny adres IP twojego hosta i wyglądać podobnie do tego:

```
myip                                            [s|192.0.2.1]
```

W ten sposób utworzyłeś prosty, ładowalny plugin dla agent 2 Zabbixa. Gratulacje!

```
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{}

    // Zarejestruj klucz pozycji `myip`.
    err := plugin.RegisterMetrics(
        p,
        "MyIP",                          // Nazwa wtyczki
        "myip",                          // Nazwa klucza pozycji
        "Zwraca adres IP hosta.",        // Opis klucza pozycji
    )
    if err != nil {
        return errs.Wrap(err, "failed to register metrics")
    }

    // Utwórz nowy program obsługi.
    h, err := container.NewHandler("MyIP") // Nazwa wtyczki
    if err != nil {
        return errs.Wrap(err, "failed to create new handler")
    }

    // Skonfiguruj logowanie, aby przekazywać logi z wtyczki do agenta.
    // Dostępne przez p.Logger.Infof, p.Logger.Debugf itd.
    p.Logger = h

    // Uruchom wykonywanie wtyczki.
    // Blokuje do momentu otrzymania żądania zakończenia od agenta.
    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) {
    // Wtyczka może używać różnej logiki zbierania danych w zależności od parametru `key`.
    // Ta implementacja jedynie sprawdza, czy podany `key` jest obsługiwany.
    if key != "myip" {
        return nil, errs.Errorf("unknown item key %q", key)
    }

    // Log zostanie przekazany do logu agenta 2.
    p.Infof(
        "received request to handle %q key with %d parameters",
        key,
        len(params),
    )

    // Zbierz dane i zwróć je.

    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
}
```

Pełny kod źródłowy

Interfejsy wtyczek

Ta sekcja opisuje dostępne interfejsy wtyczek.

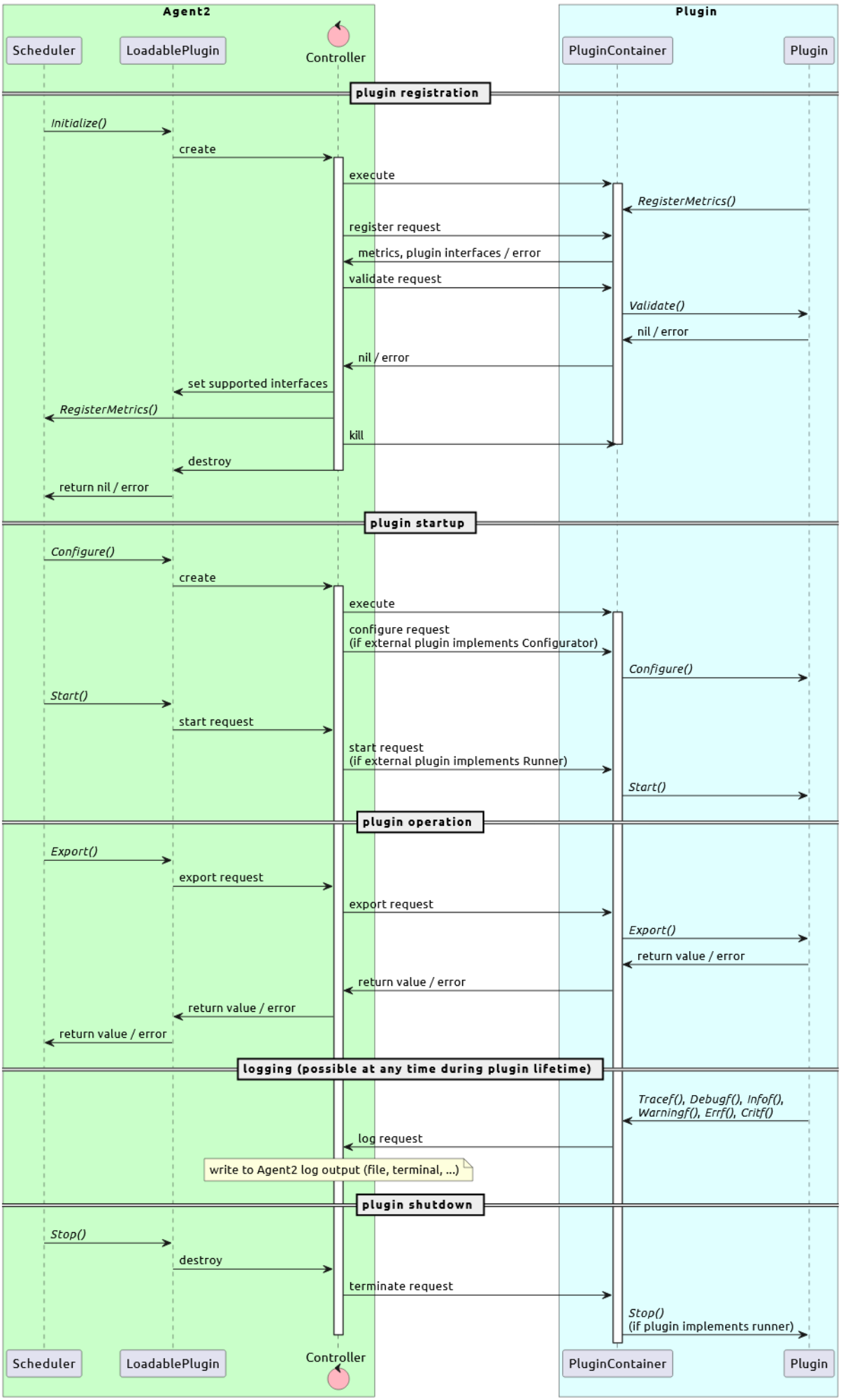
plugin.Exporter *Exporter* to najprostszy interfejs, który wykonuje badanie, zwraca wartość (wartości), nic lub błąd. Akceptuje przygotowany klucz elementu, parametry i kontekst. Dostęp do wszystkich innych interfejsów wtyczek jest wyłączny i żadna metoda nie może zostać wywołana, jeśli wtyczka wykonuje już zadanie. Ponadto istnieje limit maksymalnie 100 równoczesnych wywołań *Export()* dla wtyczki, który można zmniejszyć zgodnie z wymaganiami dla każdej wtyczki.

plugin.Configurator Interfejs *Configurator* udostępnia parametry konfiguracji wtyczki z plików konfiguracyjnych agenta Zabbix 2.

plugin.Runner Interfejs *Runner* zapewnia możliwości do wykonywania inicjalizacji, gdy wtyczka jest uruchamiana (aktywowana) i deinicjalizacji, gdy wtyczka jest zatrzymywana (dezaktywowana). Na przykład wtyczka może uruchomić/zatrzymać jakąś w tle *goroutine*, implementując interfejs Runner.

Diagram połączenia Zabbix agent 2 łączy się dwukierunkowo z wtyczkami za pomocą gniazd UNIX w systemie Linux oraz potoków nazwanych w systemie Windows.

Poniższy diagram połączenia ilustruje proces komunikacji między Zabbix agent 2 a ładowalną wtyczką oraz proces zbierania metryk.


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

Biblioteka Python dla Zabbix

zabbix_utils to oficjalna biblioteka Pythona dla Zabbix.

Za pomocą zabbix_utils możesz:

- Używać API Zabbix do zarządzania obiektami Zabbix (tworzyć hosty, aktualizować pozycje, pobierać zdarzenia itp.).
- Zbierać dane z agent Zabbix (podobnie jak za pomocą Zabbix get).
- Wysyłać dane do serwer Zabbix lub proxy (podobnie jak za pomocą Zabbix sender).

Biblioteka obsługuje zarówno synchroniczne, jak i asynchroniczne wykonywanie skryptów, co pozwala jej obsługiwać zarówno proste przepływy pracy, jak i zakrojone na szeroką skalę operacje równoległe, takie jak masowe zbieranie danych, eksport zbiorczy, przetwarzanie alertów oraz efektywne skalowanie systemu.

Biblioteka jest zgodna z obsługiwanymi wersjami serwer Zabbix oraz nowoczesnymi wydaniami Pythona.

Aby rozpocząć, zainstaluj bibliotekę, a następnie skorzystaj z przewodnika szybki start.

Instalacja

zabbix_utils można zainstalować przy użyciu dowolnej z metod opisanych poniżej.

Wymagania:

- Zabbix 6.0 lub nowszy (testowano na 6.0, 7.0, 7.2, 7.4)
- Python 3.8 lub nowszy (testowano na 3.8–3.13)
- asynchroniczny framework HTTP aiohttp (wymagany tylko w trybie asynchronicznym)

Python Package Index (PyPI)

Jest to najczęściej stosowana metoda w większości środowisk:

```
pip install zabbix_utils
pip install zabbix_utils[async] # Tylko dla obsługi async
```

Repozytorium Zabbix

Użyj tej metody, jeśli wolisz zarządzać zależnościami za pomocą menedżera pakietów swojego systemu.

Pobierz pakiety Zabbix dla swojej dystrybucji Linuksa, a następnie uruchom poniższe polecenia.

W systemie RHEL i pochodnych:

```
dnf install python3-zabbix-utils
dnf install epel-release # Tylko dla obsługi async
dnf install python3-aiohttp # Tylko dla obsługi async
```

W systemach Debian/Ubuntu i pochodnych:

```
apt install python3-zabbix-utils
apt install python3-aiohttp # Tylko dla obsługi async
```

Źródło (GitHub)

Użyj tej metody, jeśli preferujesz najnowszą wersję rozwojową:

```
git clone https://github.com/zabbix/python-zabbix-utils
cd python-zabbix-utils/
pip install -r requirements.txt # Tylko dla obsługi asynchronicznej
python3 setup.py install
```

Aby zainstalować zależności wymagane dla trybu asynchronicznego, możesz także użyć jednej z metod opisanych powyżej.

Przewodnik szybkiego startu

Po zainstalowaniu zabbix_utils możesz używać go w swoim skrypcie.

Ten przewodnik szybkiego startu pokazuje, jak:

- Pobierać nazwę hosta z Zabbix agent.
- Tworzyć host i pozycja typu trapper za pomocą Zabbix API.
- Wysyłać wartość do pozycji.

Przewodnik przedstawia skrypt krok po kroku, wyjaśniając każdą część w miarę jej wprowadzania. Na końcu strony znajduje się kompletny skrypt.

W przewodniku założono również, że Twój serwer Zabbix, agent i API działają lokalnie.

Pobieranie nazwy hosta z Zabbix agent

Zacznij od pobrania nazwy hosta systemu, na którym działa Zabbix agent. Ta nazwa hosta będzie potrzebna do utworzenia hosta przy użyciu Zabbix API.

1. Zaimportuj klasę `Getter` z zabbix_utils. Ta klasa działa podobnie jak Zabbix get i umożliwia żądanie danych z Zabbix agent.

2. Utwórz instancję `Getter`, która łączy się z lokalnym Zabbix agent pod adresem `127.0.0.1` na porcie `10050`.

3. Wywołaj metodę `get()` na instancji `Getter`, aby zażądać nazwy hosta z Zabbix agent.

- Metoda `get()` przyjmuje klucz pozycji jako parametr i wysyła żądanie do Zabbix agent o tę pozycję.
- Metoda `get()` zwraca obiekt, a nazwa hosta jest przechowywana w atrybucie `value` tego obiektu.
- `system.hostname` to wbudowany klucz Zabbix agent, który zwraca nazwę hosta.

```
from zabbix_utils import Getter

agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value
```

Połącz się i zaloguj do Zabbix API

Następnie połącz się z Zabbix API. Pozwoli to Twojemu skryptowi zarządzać obiektami Zabbix, takimi jak hosty i pozycje.

1. Zaimportuj klasę `ZabbixAPI` z zabbix_utils.

2. Utwórz instancję `ZabbixAPI` i podaj adres URL swojego interfejsu webowego Zabbix.

3. Wywołaj metodę `login()` na instancji `ZabbixAPI` i podaj swoją nazwę użytkownika oraz hasło. Konto musi mieć uprawnienie do uzyskiwania dostępu do Zabbix API.

4. Wywołaj metodę `logout()`, aby zamknąć sesję po zakończeniu operacji API.

```
from zabbix_utils import ZabbixAPI

api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

### API operations go here

api.logout()
```

Utwórz host w Zabbix

Teraz, gdy masz nazwę hosta z Zabbix agent i jesteś połączony z Zabbix API, możesz utworzyć nowy host w Zabbix.

1. Wywołaj metodę API `host.create()` na instancji `ZabbixAPI` i podaj szczegóły hosta:

- `host` — ustaw na zmienną `hostname` zawierającą nazwę hosta pobraną z Zabbix agent.
- `interfaces` — zawiera szczegóły połączenia dla Zabbix agent uruchomionego na hoście.
- `groups` — zawiera co najmniej jedną grupę hostów, do której host powinien należeć.

2. Ponieważ Zabbix API zwraca ID nowo utworzonego hosta, zapisz to ID w zmiennej `host_id` do późniejszego użycia.

3. Wyświetl komunikat potwierdzający, że host został utworzony.

```
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}' created with ID {host_id}")
```

Utwórz nową pozycję w Zabbix

Po utworzeniu hosta możesz dodać do niego pozycję.

1. Zdefiniuj unikalny klucz dla pozycji, która ma zostać utworzona.

2. Wywołaj metodę `item.create()` na instancji `ZabbixAPI` i podaj szczegóły pozycji:

- `hostid` - ustaw na zmienną `host_id` zawierającą ID hosta, który został właśnie utworzony.
- `name` - nazwa pozycji.
- `key_` - ustaw na zmienną `item_key`, którą właśnie zdefiniowano.
- `type` - ustaw na 2 (pozycja trapper); jest to wymagane do odbierania wartości wysyłanych ze skryptu w kolejnych krokach.
- `value_type` - ustaw na 3 (numeric unsigned), czyli typ danych przechowywanych przez tę pozycję.

3. Ponieważ Zabbix API zwraca ID nowo utworzonej pozycji, zapisz to ID w zmiennej `item_id` do późniejszego użycia.

4. Wyświetl komunikat potwierdzający utworzenie pozycji.

```
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}' created with ID {item_id}")
```

Wyślij wartość do hosta

Teraz, gdy utworzyłeś host i pozycję, możesz wysłać do niej dane.

1. Zaimportuj klasę `time` i odczekaj kilka sekund przed wysłaniem danych. Dzięki temu Zabbix w pełni przetworzy nowy host i pozycję.

2. Zaimportuj klasę `Sender` z zabbix_utils. Ta klasa działa podobnie jak Zabbix sender i umożliwia skryptowi wysyłanie danych do Zabbix.

3. Utwórz instancję `Sender`, która łączy się z lokalnym serwerem Zabbix pod adresem `127.0.0.1` na porcie `10051`.

4. Wywołaj metodę `send_value()` na instancji `Sender`, aby wysłać wartość do pozycji hosta, i podaj następujące szczegóły:

- `hostname` — ustaw na zmienną `hostname` zawierającą nazwę hosta pobraną z Zabbix agent.
- `item_key` — ustaw na zmienną `item_key`, którą właśnie wcześniej zdefiniowałeś w skrypcie.
- `1` — wartość do wysłania.

4. Wyświetl komunikat potwierdzający, że wartość została wysłana.

```
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}")
```

Po pomyślnym wykonaniu tego kodu powinien zostać wyświetlony komunikat o powodzeniu informujący, że Zabbix odebrał wartość:

```
Sender response: {"processed": 1, "failed": 0, "total": 1, "time": "0.000151", "chunk": 1}
```

Teraz możesz również sprawdzić wartość w interfejsie webowym Zabbix (*Monitoring* > *Latest data*).

Kompletny skrypt

Poniżej znajduje się kompletny skrypt, który łączy wszystkie kroki: pobranie nazwy hosta, utworzenie hosta i pozycji przy użyciu Zabbix API oraz wysłanie danych do Zabbix.

```
import time
from zabbix_utils import Getter, ZabbixAPI, Sender

### Pobierz nazwę hosta z Zabbix agent
agent = Getter(host='127.0.0.1', port=10050)
hostname = agent.get('system.hostname').value

### Połącz z Zabbix API i zaloguj się
api = ZabbixAPI(url='127.0.0.1/zabbix')
api.login(user='Admin', password='zabbix')

### Utwórz host w 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}' created with ID {host_id}")

### Utwórz pozycję w 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}' created with ID {item_id}")

### Wyloguj się z API
api.logout()

### Poczekaj, aż Zabbix przetworzy nowy host
time.sleep(10)

### Wyślij wartość do hosta
sender = Sender(server='127.0.0.1', port=10051)
```

```
response = sender.send_value(hostname, item_key, 1)

print(f"Sender response: {response}")
```

Korzystanie z API Zabbix

zabbix_utils umożliwia korzystanie z API Zabbix do zarządzania obiektami Zabbix, w tym tworzenia hostów, aktualizowania pozycji, pobierania zdarzeń i nie tylko.

Żądania API mogą być wykonywane w trybie synchronicznym lub asynchronicznym:

- W trybie synchronicznym skrypt Python wysyła żądanie i czeka na odpowiedź przed kontynuowaniem działania, co jest odpowiednie dla prostych, sekwencyjnych i przewidywalnych operacji.
- W trybie asynchronicznym skrypt wysyła żądania bez oczekiwania na każdą odpowiedź, umożliwiając równoległe wykonywanie innych operacji; jest to bardziej efektywne w przypadku wolnych żądań lub dużych partii danych.

Przykłady na tej stronie koncentrują się na trybie synchronicznym, chociaż tryb asynchroniczny opiera się na podobnych wzorcach. Dodatkowe przykłady są dostępne w repozytorium GitHub zabbix_utils.

Import

Aby używać zabbix_utils do pracy z API Zabbix, zaimportuj klasę `ZabbixAPI` w swoim skrypcie:

```
from zabbix_utils import ZabbixAPI
```

> **Note:**
> Jeśli już używasz biblioteki Python społeczności, zazwyczaj możesz zastąpić ten import przez `ZabbixAPI` z zabbix_utils.

Logowanie

Przed wykonywaniem żądań API należy utworzyć instancję `ZabbixAPI` i zalogować się do Zabbix API.

Wybierz metodę, która najlepiej odpowiada sposobowi zarządzania poświadczeniami. Możesz użyć nazwy użytkownika i hasła lub tokenów API.

Podczas inicjalizacji

Podczas tworzenia instancji `ZabbixAPI` możesz od razu podać adres URL interfejsu webowego Zabbix oraz swoje dane uwierzytelniające:

```
### Nazwa użytkownika i hasło:
api = ZabbixAPI(url="127.0.0.1/zabbix", user="Admin", password="zabbix")

### Token API:
api = ZabbixAPI(url="127.0.0.1/zabbix", token="your_api_token")
```

Jeśli wolisz pogrupować parametry połączenia, możesz przekazać je jako słownik Pythona i rozpakować go za pomocą składni argumentów nazwanych Pythona:

```
ZABBIX_AUTH = {
    "url": "127.0.0.1/zabbix",
    "user": "Admin",
    "password": "zabbix"
}

api = ZabbixAPI(**ZABBIX_AUTH)
```

Używanie `login()`

Możesz najpierw utworzyć instancję `ZabbixAPI`, podając tylko URL interfejsu webowego Zabbix, a następnie użyć metody `login()` ze swoimi poświadczeniami:

```
### Nazwa użytkownika i hasło:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### Token API:
```

```
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(token="your_api_token")
```

Używanie zmiennych środowiskowych

Możesz najpierw zapisać swoje dane uwierzytelniające jako zmienne środowiskowe, na przykład przy użyciu terminala:

```
### Nazwa użytkownika i hasło:
export ZABBIX_USER="Admin"
export ZABBIX_PASSWORD="zabbix"

### Token:
export ZABBIX_TOKEN="your_api_token"
```

Następnie, jeśli uruchomisz swój skrypt z tej samej sesji terminala, instancja `ZabbixAPI` w skrypcie wymaga jedynie adresu URL interfejsu webowego Zabbix:

```
api = ZabbixAPI(url="127.0.0.1/zabbix")
```

Możesz również zapisać adres URL w zmiennej środowiskowej:

```
### Nazwa użytkownika i hasło:
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"
```

Gdy adres URL i dane uwierzytelniające są już ustawione jako zmienne środowiskowe, instancja `ZabbixAPI` w skrypcie nie wymaga już żadnych argumentów:

```
api = ZabbixAPI()
```

Wylogowanie

Jeśli zalogowano się przy użyciu nazwy użytkownika i hasła, po zakończeniu operacji API należy wywołać metodę `logout()`:

```
from zabbix_utils import ZabbixAPI

api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### API actions go here

api.logout()
```

> **Note:**
> Wywołanie metody `logout()` nie jest wymagane podczas używania tokenów API.

Żądania API

Po zalogowaniu możesz wykonywać dowolne żądania API, wywołując metody opisane w opisie metod API Zabbix.

Metody API są wywoływane przy użyciu następującego formatu:

```
api_instance.zabbix_object.method(parameters)
```

> **Note:**
> Niektóre nazwy metod lub obiektów API Zabbix używają słów, które są słowami zastrzeżonymi w Pythonie (np. `import`). Aby uniknąć błędów Pythona, podczas używania ich w Pythonie dodaj podkreślenie do nazwy metody lub obiektu (np. `api.configuration.import_`).

Na przykład, aby wyświetlić listę hostów za pomocą `host.get`:

```
### 1. Importuj ZabbixAPI z zabbix_utils:
from zabbix_utils import ZabbixAPI
```

```
### 2. Utwórz instancję ZabbixAPI i zaloguj się:
api = ZabbixAPI(url="127.0.0.1/zabbix")
api.login(user="Admin", password="zabbix")

### 3. Pobierz hosty pasujące do filtra:
hosts = api.host.get(
    filter={
        "host": [
            "Zabbix server",
            "Linux server"
        ]
    }
)

### 4. Iteruj po zwróconych hostach i wyświetl szczegóły każdego hosta:
for host in hosts:
    print(host)

### 5. Wyloguj się z API, aby zamknąć sesję:
api.logout()
```

Tryb asynchroniczny

Tryb asynchroniczny pozwala skryptowi wysyłać żądania API bez czekania na zakończenie każdego z nich. Może to zwiększyć wydajność skryptu, gdy musi on wykonać wiele żądań API lub gdy niektóre żądania zajmują dużo czasu.

Podczas korzystania z trybu asynchronicznego występują istotne różnice w porównaniu z trybem synchronicznym:

- Zaimportuj moduł `asyncio` języka Python (należy najpierw zainstalować wymagane zależności).
- Zaimportuj `AsyncZabbixAPI` zamiast `ZabbixAPI`.
- Napisz kod wewnątrz funkcji `async`.
- Używaj `await` podczas wywoływania metod API, w tym `login()` i `logout()`.

Attention:
Podczas tworzenia instancji `AsyncZabbixAPI` nie można zalogować się podczas inicjalizacji.

Na przykład, aby wyświetlić listę hostów za pomocą `host.get` w trybie asynchronicznym:

```
### 1. Zaimportuj asyncio dla trybu asynchronicznego oraz AsyncZabbixAPI z zabbix_utils:
import asyncio
from zabbix_utils import AsyncZabbixAPI

### 2. Zdefiniuj główną funkcję async, w której będą wykonywane wszystkie operacje API:
async def main():

    # 3. Utwórz instancję AsyncZabbixAPI i zaloguj się (wymaga await):
    api = AsyncZabbixAPI(url="127.0.0.1")
    await api.login(user="User", password="zabbix")

    # 4. Pobierz hosty pasujące do filtra (wymaga await):
    hosts = await api.host.get(
        filter={
            "host": [
                "Zabbix server",
                "Linux server"
            ]
        }
    )

    # 5. Iteruj po zwróconych hostach i wypisz szczegóły każdego hosta:
    for host in hosts:
        print(host)

    # 6. Wyloguj się z API, aby zamknąć sesję (wymaga await):
```

```
    await api.logout()

### 7. Uruchom główną funkcję async main() przy użyciu pętli zdarzeń asyncio:
asyncio.run(main())
```

Przykłady

Poniższe przykłady pokazują typowe zadania API Zabbix z użyciem biblioteki zabbix_utils.

Dodatkowe przykłady są dostępne w katalogu `examples` repozytorium GitHub zabbix_utils.

Zbieranie danych z agenta Zabbix

Przegląd

zabbix_utils umożliwia zbieranie danych z Zabbix agent (podobnie jak Zabbix get).

Dane mogą być zbierane w trybie synchronicznym lub asynchronicznym:

- W trybie synchronicznym skrypt Python wysyła żądanie i czeka na dane przed kontynuowaniem działania, co jest odpowiednie dla prostych, sekwencyjnych i przewidywalnych operacji.
- W trybie asynchronicznym skrypt wysyła żądania bez oczekiwania na każdą odpowiedź, umożliwiając równoległe wykonywanie innych operacji; jest to bardziej wydajne w przypadku wolnych żądań lub dużych partii danych.

Przykłady na tej stronie koncentrują się na trybie synchronicznym, chociaż tryb asynchroniczny opiera się na podobnych wzorcach. Dodatkowe przykłady są dostępne w repozytorium GitHub zabbix_utils.

Import

Aby używać `zabbix_utils` do zbierania wartości pozycji, zaimportuj klasę `Getter` w swoim skrypcie Python:

```
from zabbix_utils import Getter
```

Dane żądania

Aby zażądać wartości pozycji:

1. Utwórz instancję `Getter`, określając adres IP i port swojego agenta Zabbix.
2. Wywołaj metodę `get()` na instancji `Getter`, określając klucz pozycji, którą chcesz pobrać.

Na przykład, aby zażądać danych dla pozycji `system.uname`:

```
agent = Getter(host='192.0.2.1', port=10050)
response = agent.get('system.uname')
```

Używanie niestandardowego IP

Jeśli serwer, na którym uruchamiany jest skrypt, ma wiele adresów IP, możesz określić `source_ip`, którego `Getter` ma używać podczas łączenia się z agentem Zabbix:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    source_ip='10.10.7.1'
)
```

Używanie timeoutu

Możesz ustawić `timeout` odpowiedzi dla `Getter`, aby kontrolować, jak długo skrypt ma czekać na odpowiedź od agent przed rezygnacją:

```
agent = Getter(
    host='192.0.2.1',
    port=10050,
    timeout=30
)
```

Używanie szyfrowania

`Getter` nie zawiera wbudowanej obsługi szyfrowania, ale można ją zapewnić, tworząc wrapper z użyciem bibliotek firm trzecich:

```
def psk_wrapper(sock, tls):
    # ...
    # Implementation of TLS PSK wrapper for the socket
    # ...

agent = Getter(
    host='192.0.2.1',
    port=10050,
    socket_wrapper=psk_wrapper
)
```

Odpowiedź

Odpowiedź z Zabbix agent jest przetwarzana przez bibliotekę i zwracana jako obiekt `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
```

Tryb asynchroniczny

Tryb asynchroniczny pozwala skryptowi zbierać wartości bez czekania na nadejście każdej z nich. Może to zwiększyć wydajność skryptu, gdy musi on zebrać wiele wartości lub gdy niektóre wartości są zbierane przez dłuższy czas.

Podczas korzystania z trybu asynchronicznego występują ważne różnice w porównaniu z trybem synchronicznym:

- Zaimportuj moduł Python `asyncio` (najpierw musisz zainstalować wymagane zależności).
- Zaimportuj `AsyncGetter` zamiast `Getter`.
- Napisz kod wewnątrz funkcji `async`.
- Użyj `await` podczas wywoływania metody `get()`.

Na przykład, aby zebrać pojedynczą wartość przy użyciu trybu asynchronicznego:

```
### 1. Import asyncio for asynchronous mode, and AsyncGetter from zabbix_utils:
import asyncio
from zabbix_utils import AsyncGetter

### 2. Define the main async function where all data requests will be executed:
async def main():
    agent = AsyncGetter(host='192.0.2.1', port=10050)

    # 3. Fetch the system.uname value from Zabbix agent (must await):
    response = await agent.get('system.uname')

    # 4. Print the value returned by Zabbix agent:
    print(response.value)

### 5. Run the async main() function using asyncio's event loop:
asyncio.run(main())
```

Wysyłanie danych do serwera lub proxy Zabbix

zabbix_utils umożliwia wysyłanie wartości pozycji do pozycji trapper na serwer lub proxy Zabbix (podobnie jak Zabbix sender).

Możesz wysłać pojedynczą wartość, wiele wartości, a nawet kierować dane do wielu klastrów Zabbix.

Dane mogą być wysyłane w trybie synchronicznym lub asynchronicznym:

- W trybie synchronicznym skrypt Python wysyła wartości i czeka na odpowiedź przed kontynuowaniem działania; jest to odpowiednie rozwiązanie dla prostych, sekwencyjnych i przewidywalnych operacji.
- W trybie asynchronicznym skrypt wysyła wartości bez oczekiwania na każdą odpowiedź, umożliwiając równoległe wykonywanie innych operacji; jest to bardziej efektywne w przypadku wolnych żądań lub dużych partii danych.

Przykłady na tej stronie koncentrują się na trybie synchronicznym, chociaż tryb asynchroniczny opiera się na podobnych wzorcach. Dodatkowe przykłady są dostępne w repozytorium GitHub zabbix_utils.

Import

Aby używać `zabbix_utils` do wysyłania wartości pozycji, zaimportuj klasę `Sender` w swoim skrypcie:

```
from zabbix_utils import Sender
```

Aby wysyłać wiele wartości, możesz również zaimportować klasę `ItemValue`:

```
from zabbix_utils import Sender, ItemValue
```

Wyślij pojedynczą wartość

Aby wysłać wartość pozycji:

1. Utwórz instancję `Sender`, podając adres IP i port swojego serwera lub proxy Zabbix.
2. Wywołaj metodę `send_value()` na instancji `Sender`, używając następującego formatu:

```
sender_instance.send_value('host', 'item.key', 'value', optional_timestamp, optional_nanoseconds)
```

Na przykład, aby wysłać 1 do pozycji trapper `service.status` na hoście `Linux server`:

```
sender = Sender(server='127.0.0.1', port=10051)
response = sender.send_value('Linux server', 'service.status', 1)
```

Używanie niestandardowego adresu IP

Jeśli serwer uruchamiający skrypt ma wiele adresów IP, możesz określić `source_ip`, którego `Sender` ma używać podczas wysyłania wartości do serwera Zabbix lub proxy:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    source_ip='10.10.7.1'
)
```

Używanie timeout

Możesz ustawić `timeout` odpowiedzi dla `Sender`, aby kontrolować, jak długo skrypt powinien czekać na odpowiedź z serwera Zabbix lub proxy, zanim zrezygnuje:

```
sender = Sender(
    server='127.0.0.1',
    port=10051,
    timeout=30
)
```

Korzystanie z pliku konfiguracyjnego agenta

Możesz zezwolić bibliotece `zabbix_utils` na odczytywanie parametrów `Server` lub `ServerActive` z lokalnego pliku konfiguracyjnego Zabbix agenta lub agenta 2. W takim przypadku nie musisz określać parametrów połączenia podczas tworzenia instancji `Sender`:

```
sender = Sender(
    use_config=True,
    config_path='/etc/zabbix/zabbix_agent2.conf'
)
```

Attention:
Jeśli parametr `ServerActive` zawiera co najmniej jeden klaster Zabbix z wieloma instancjami serwera, `Sender` wysyła dane do pierwszego dostępnego serwera w każdym klastrze. Jeśli parametr `ServerActive` nie jest ustawiony, używany jest adres z parametru `Server` z domyślnym portem (10051).

Używanie szyfrowania

Sender nie zawiera wbudowanej obsługi szyfrowania, ale można ją zapewnić, tworząc opakowanie przy użyciu bibliotek zewnętrznych:

```
def psk_wrapper(sock, tls):
    # ...
    # Implementacja opakowania TLS PSK dla gniazda
    # ...

sender = Sender(
    server='127.0.0.1',
    port=10051,
    socket_wrapper=psk_wrapper
)
```

Odpowiedź dla pojedynczej wartości

Odpowiedź zwrócona przez serwer Zabbix lub proxy jest przetwarzana przez bibliotekę i zwracana jako obiekt 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
```

Tryb asynchroniczny

Tryb asynchroniczny pozwala skryptowi Python wysyłać wartości bez oczekiwania na odpowiedź z serwera Zabbix lub proxy. Może to zwiększyć wydajność skryptu, gdy musi on wysłać wiele wartości albo gdy wysłanie niektórych wartości zajmuje dużo czasu.

Podczas korzystania z trybu asynchronicznego występuje kilka istotnych różnic w porównaniu z trybem synchronicznym:

- Zaimportuj moduł asyncio języka Python (najpierw należy zainstalować wymagane zależności).
- Zaimportuj AsyncSender zamiast Sender.
- Napisz kod wewnątrz funkcji async.
- Użyj await podczas wywoływania metody send_value().

Na przykład, aby wysłać pojedynczą wartość w trybie asynchronicznym:

```
### 1. Import asyncio for asynchronous mode, and AsyncSender from zabbix_utils:
import asyncio
from zabbix_utils import AsyncSender

### 2. Define the main async function where all data sending operations (must await) will be executed:
async def main():
    sender = AsyncSender(server='127.0.0.1', port=10051)
    response = await sender.send_value('Linux server', 'service.status', 1)

    # 3. Print the response returned by Zabbix server or proxy:
    print(response)

### 4. Run the async main() function using asyncio's event loop:
asyncio.run(main())
```

Wysyłanie wielu wartości

Aby wysłać wiele wartości:

1. Przygotuj tablicę obiektów ItemValue, z których każdy używa tego samego formatu co metoda send_value().
2. Utwórz instancję Sender, podając adres IP i port serwera lub proxy Zabbix.
3. Wywołaj metodę send() (zamiast send_value()) na instancji Sender, przekazując tablicę obiektów z wartościami do wysłania.

Na przykład, aby wysłać pięć wartości do różnych hostów:

```
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)
```

Używanie niestandardowego rozmiaru fragmentu

Jeśli chcesz wysłać więcej wartości, niż pozycja trapper może przyjąć w pojedynczym żądaniu, możesz podzielić je na fragmenty.

Domyślnie rozmiar fragmentu wynosi 250 wartości. Możesz go zmienić, ustawiając parametr `chunk_size` podczas tworzenia instancji `Sender`.

Na przykład, aby wysłać pięć wartości w trzech fragmentach (2-2-1), ustaw parametr `chunk_size` na 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)
```

Wysyłanie wartości do wielu klastrów Zabbix

Aby wysyłać wartości do wielu klastrów Zabbix:

1. Przygotuj tablicę klastrów Zabbix. Jeśli klaster ma wiele węzłów, wartość zostanie wysłana do pierwszego **dostępnego** węzła każdego klastra.
2. Utwórz obiekt `Sender`, podając tablicę klastrów Zabbix.
3. Wywołaj metodę `send_value()` na instancji `Sender`, używając tego samego formatu jak w metodzie `send_value()`.

Na przykład, aby wysłać wartość do pierwszego dostępnego węzła w każdym klastrze:

```
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)
```

Odpowiedź dla wielu wartości

Domyślnie `Sender` zwraca zagregowany wynik wysyłania wartości dla wszystkich hostów lub klastrów:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}
```

Jeśli potrzebujesz bardziej szczegółowych informacji, możesz sprawdzić wyniki dla każdego klastra i każdego fragmentu za pomocą atrybutu `response.details`:

```
print(response)
### {"processed": 2, "failed": 0, "total": 2, "time": "0.000108", "chunk": 2}

if response.failed == 0:
    print(f"Wartość została wysłana pomyślnie w {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"Przetworzono {resp.processed} z {resp.total} na {node.address}:{node.port}")
            # Przetworzono 1 z 1 na 127.0.0.1:10051
            # Przetworzono 1 z 1 na zabbix.example.local:10051
```

Rejestrowanie debugowania

Aby rozwiązywać problemy podczas pracy z Zabbix API, zbierania danych z Zabbix agent lub wysyłania danych do Zabbix serwer lub proxy, można włączyć rejestrowanie debugowania.

Biblioteka zabbix_utils używa standardowego modułu `logging` języka Python.

Aby wyświetlać szczegółowe komunikaty debugowania z biblioteki:

1. Zaimportuj do skryptu moduł `logging` języka Python.
2. Skonfiguruj system rejestrowania tak, aby wyprowadzał komunikaty w czytelnym formacie, i ustaw poziom rejestrowania na 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)
```

Po włączeniu rejestrowania debugowania wyświetlane są szczegółowe informacje o żądaniach i odpowiedziach, na przykład:

```
[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
```

Moduły

Co to jest moduł frontendowy PHP?

- Moduł to jednostka posiadająca unikalny identyfikator, nazwę, opis, autora i inne pola zdefiniowane w pliku manifestu, wraz z PHP, JavaScript i innymi plikami znajdującymi się w jednym katalogu w katalogu *modules* instalacji frontendu Zabbix (na przykład *zabbix/ui/modules*).
- Moduł powinien spełniać proste zasady aby zapewnić poprawne działanie.
- Moduł musi zostać zainstalowany (rozpakowany) i włączony w interfejsie frontendu przez administratora.

Do czego można wykorzystać moduł

- Dodawanie nowych funkcjonalności poprzez niestandardowe sekcje frontendu;
- Tworzenie niestandardowych typów widgetów dashboardu (zobacz moduły widgetów);
- Zastąpienie lub rozszerzenie istniejącej funkcjonalności.

Do czego nie można wykorzystać modułu

- Rejestracja nowej metody API lub modyfikacja istniejącej.

Jak działają moduły

- Włączony moduł jest uruchamiany przy każdym żądaniu HTTP, przed wykonaniem kodu akcji.
- Moduł zarejestruje nowe akcje lub zdefiniuje na nowo istniejące.
- Moduł doda nowe sekcje front-endu i usunie lub zdefiniuje na nowo istniejące.
- Moduł będzie podłączał się do zdarzeń front-endu, takich jak onBeforeAction i onTerminate, jeśli będzie to konieczne.
- Żądana akcja jest ostatecznie wykonywana przez uruchomienie kodu akcji - domyślnego lub zdefiniowanego w module.

Gdzie dalej Niezależnie od tego, czy wolisz uczyć się poprzez działanie, czy najpierw przeczytać wytyczne, te strony zawierają informacje i kroki wymagane do zbudowania własnych modułów:

- Samouczki krok po kroku dotyczące pisania pierwszego modułu
- Struktura pliku modułu
- Szczegóły modułu widżetu
- Przykłady modułów do ponownego wykorzystania

Struktura plików modułu

Cały kod związany z modułem jest przechowywany w pojedynczym katalogu wewnątrz katalogu **modules** instalacji frontendu Zabbix (na przykład *zabbix/ui/modules*).

```
example_module_directory/ (wymagane)
   manifest.json (wymagane) Metadane i definicja akcji.
   Module.php Inicjalizacja modułu i obsługa zdarzeń.
   actions/   pliki kontrolera akcji.
       SomethingView.php
       SomethingCreate.php
       SomethingDelete.php
       data_export/
           ExportAsXml.php
           ExportAsExcel.php
   views/   pliki widoków.
       example.something.view.php
       example.something.delete.php
   assets/   wszelkie dodatkowe pliki do wykorzystania w widokach. Należy określić w pliku manifest.json.
       js/   skrypty JavaScript używane w widokach.
           example.something.view.js.php
       css/   pliki CSS używane w widokach.
           example.something.css
```

```
        image.png    obraz do użycia w widokach.
        example.something.file    dowolny plik do użycia w widokach.
```

Drzewo plików modułów

Pisanie modułu Przykładowy proces tworzenia modułu składa się z następujących kroków (jeśli to możliwe, kliknij nazwę pliku lub folderu, aby wyświetlić dodatkowe szczegóły dotyczące danego kroku):

1. Utwórz nowy katalog dla modułu w **zabbix/ui/modules/**.
2. Dodaj plik manifest.json z metadanymi modułu.
3. Utwórz folder *views* i zdefiniuj widok(i) modułu.
4. Utwórz folder *actions* i zdefiniuj akcję(i) modułu.
5. Utwórz plik Module.php (lub Widget.php dla widgetów pulpitu) i zdefiniuj reguły inicjalizacji oraz obsługi zdarzeń.
6. Utwórz folder *assets* na pliki JavaScript (umieszczane w *assets/js*), style CSS (umieszczane w *assets/css*) lub inne dodatkowe pliki.
7. Upewnij się, że w pliku manifest.json określono wymagane widoki, akcje i pliki zasobów.
8. Zarejestruj moduł we frontendzie Zabbixa i zacznij z niego korzystać.

Note:
Możesz zarejestrować i włączyć moduł zaraz po utworzeniu pliku manifest.json. Po włączeniu modułu możesz natychmiast podejrzeć wszystkie zmiany wprowadzone w plikach modułu, odświeżając frontend Zabbixa.

manifest.json

Każdy moduł potrzebuje pliku manifest.json. Plik musi znajdować się w głównym katalogu modułu (na przykład *zabbix/ui/modules/module_name/manifest.json*).

Minimalnie plik manifest.json powinien określać następujące pola:

```
{
    "manifest_version": 2.0,
    "id": "my_ip_address",
    "name": "My IP Address",
    "namespace": "MyIPAddress",
    "version": "1.0"
}
```

Parametry obsługiwane w **manifest.json** (kliknij nazwę parametru, aby wyświetlić szczegółowy opis):

Parameter	Description	Required
manifest_version	Wersja manifestu modułu.	Yes
id	Unikalny identyfikator modułu.	
name	Nazwa modułu, która będzie wyświetlana w sekcji Administracja.	
namespace	Przestrzeń nazw PHP dla klas modułu.	
version	Wersja modułu.	
type	Typ modułu. Dla widgetu musi być ustawione na *widget*	Yes for widgets, otherwise no
widget	Konfiguracja widgetu. Używana tylko dla widgetów.	
actions	Akcje do zarejestrowania w module.	
assets	Arkusze stylów CSS i pliki JavaScript do dołączenia.	No
author	Autor modułu.	
config	Domyślne wartości niestandardowych opcji modułu.	
description	Opis modułu.	
url	Link do opisu modułu.	

manifest_version

Wersja manifestu modułu. Obecnie obsługiwana wersja to **2.0**.

Typ: Double

Przykład:

```
"manifest_version": 2.0
```

ID

Identyfikator modułu. Musi być unikalny. Aby uniknąć przyszłych konfliktów w nazewnictwie, zaleca się używanie przedrostków dla modułów (nazwa autora, firmy lub jakakolwiek inna). Na przykład, jeśli moduł jest przykładem lekcji, a nazwa modułu to ”Mój moduł”, wówczas identyfikatorem będzie ”przykład_mój_moduł”.

Typ: String

Przykład:

```
"id": "przykład_mój_moduł"
```

name

Nazwa modułu, która będzie wyświetlana w sekcji Administracja.

Typ: String

Przykład:

```
"name": "My module"
```

namespace

Przestrzeń nazw PHP dla klas modułów.

Typ: String

Przykład:

```
"namespace": "ClockWidget"
```

version

Wersja modułu. Wersja zostanie wyświetlona w sekcji Administracja.

Typ: String

Przykład:

```
"version": "1.0"
```

type

Typ modułu. Wymagane w przypadku widgetów i musi być równe ”widget”.

Typ: String

Domyślne: ”module”

Przykład:

```
"type": "widget"
```

działania

Akcje do zarejestrowania w module. Wymagane jest zdefiniowanie klucza obiektu *class* dla każdej akcji, pozostałe klucze akcji są opcjonalne.

Typ: Obiekt

Obsługiwane klucze obiektu, jeśli *typ* to *module*:

- **write.your.action.name** (obiekt) - nazwa akcji, należy zapisać małymi literami [a-z], oddzielając słowa kropkami. Obsługuje klucze:
  - **class** (string; wymagane) - nazwa klasy akcji.
  - **layout** (string) - układ akcji. Obsługiwane wartości: *layout.json*, *layout.htmlpage* (domyślnie), *null*.
  - **view** (string) - widok akcji.

Przykład:

```
"actions": {
    "moduł.example.list": {
        "class": "ExampleList",
        "view": "example.list",
        "layout": "layout.htmlpage"
```

```
        }
    }
```

Obsługiwane klucze obiektu, jeśli *type* to *widget*:

- **widget.{id}.view** (object) - nazwa pliku i klasy dla widoku widgetu. Zamień **{id}** na wartość id widżetu (na przykład *widget.example_clock.view*). Obsługuje klucze:
  - **class** (string; wymagane) - nazwa klasy akcji dla trybu widoku widgetu w celu rozszerzenia domyślnej klasy CControllerDashboardWidgetView. Plik źródłowy klasy musi znajdować się w katalogu *actions*.
  - **view** (string) - widok widżetu. Musi znajdować się w katalogu *views*. Jeśli plik widoku to *widget.view.php*, co jest wartością domyślną, można pozostawić pustą wartość. Jeśli używasz innej nazwy, podaj ją tutaj.
- **widget.{id}.edit** (object) - nazwa pliku widoku konfiguracji widgetu. Zamień **{id}** na wartość id widżetu (na przykład *widget.example_clock.edit*). Obsługuje klucze:
  - **class** (string; wymagane) - nazwa klasy akcji dla trybu widoku konfiguracji widgetu. Plik źródłowy klasy musi znajdować się w katalogu *actions*.
  - **view** (string) - widok konfiguracji widgetu. Musi znajdować się w katalogu *views*. Jeśli plik widoku to *widget.edit.php*, co jest wartością domyślną, można pozostawić pustą wartość. Jeśli używasz innej nazwy, podaj ją tutaj.

Przykład:

```
"actions": {
    "widget.tophosts.view": {
        "class": "WidgetView"
    },
    "widget.tophosts.column.edit": {
        "class": "ColumnEdit",
        "view": "column.edit",
        "layout": "layout.json"
    }
}
```

assets

Style CSS i pliki JavaScript do uwzględnienia.

Typ: Object

Obsługiwane klucze obiektów:

- **css** (tablica) - pliki CSS do dołączenia. Pliki muszą znajdować się w katalogu *assets/css*.
- **js** (tablica) - pliki JavaScript do dołączenia. Pliki muszą znajdować się w katalogu *assets/js*.

Przykład:

```
"assets": {
    "css": ["widget.css"],
    "js": ["class.widget.js"]
}
```

author

Autor modułu. Autor zostanie wyświetlony w sekcji Administracja.

Typ: String

Przykład:

```
"author": "John Smith"
```

konfiguracja

Domyślne wartości opcji modułu. Obiekt może zawierać dowolne klucze niestandardowe. Jeśli podane, wartości te zostaną zapisane do bazy danych podczas rejestracji modułu. Nowe zmienne dodane później zostaną zapisane przy pierwszym wywołaniu. Po zapisaniu, wartości zmiennych można zmieniać wyłącznie bezpośrednio w bazie danych.

Typ: Obiekt

Przykład:

```
„config": {
    "username": "Administrator",
    "password": "",
```

```
    "auth_url": "https://example.com/auth"
}
```

description

Opis modułu.

Typ: String

Przykład:

```
"description": "To jest widget zegara."
```

widget

Konfiguracja widgetu. Używana, jeśli *type* jest ustawiony na *widget*.

Typ: Object

Obsługiwane klucze obiektu:

- **name** (string) - używana na liście widgetów oraz jako domyślny nagłówek. Jeśli jest pusta, zostanie użyty parametr „name” z modułu.
- **size** (object) - domyślne wymiary widgetu. Obsługuje klucze:
  - *width* (integer) - domyślna szerokość widgetu.
  - *height* (integer) - domyślna wysokość widgetu.
- **form_class** (string) - klasa z formularzem pól widgetu. Musi znajdować się w katalogu *includes*. Jeśli klasą jest *WidgetForm.php*, oczekiwana domyślnie, ten parametr może zostać pominięty. W przypadku użycia innej nazwy należy ją tutaj określić.
- **js_class** (string) - nazwa klasy JavaScript dla trybu widoku widgetu, rozszerzającej domyślną klasę CWidget. Klasa zostanie załadowana wraz z pulpitem. Plik źródłowy klasy musi znajdować się w katalogu assets/js. Zobacz także: assets.
- **use_time_selector** (boolean) - określa, czy widget wymaga selektora czasu pulpitu. Obsługiwane wartości: *true*, *false* (domyślnie).
- **refresh_rate** (integer) - częstotliwość odświeżania widgetu w sekundach (domyślnie: 60).

Przykład:

```
"widget": {
    "name": "",
    "size": {
        "width": 12,
        "height": 5
    },
    "form_class": "WidgetForm",
    "js_class": "CWidget",
    "use_time_selector": false,
    "refresh_rate": 60
}
```

url

Link do opisu modułu. W przypadku widżetów link ten zostanie otwarty po kliknięciu ikony pomocy ? w oknie *Dodaj widżet* lub *Edytuj widżet*. Jeśli parametr **url** nie zostanie określony, kliknięcie ikony pomocy otworzy ogólną stronę Widżety pulpitu nawigacyjnego.

Typ: String

Przykład:

```
"url": "http://example.com"
```

Akcje

Za ’logikę biznesową’ modułu odpowiadają akcje. Akcja zwykle składa się z controller i action view.

Moduł może:

- Wywołać akcje, które są już zdefiniowane w interfejsie frontowym Zabbix.
- Zastąpić akcje domyślne akcjami niestandardowymi.
- Zdefiniować zupełnie nowe akcje.

Aby zastąpić domyślne zachowanie akcji jakimś niestandardowym zachowaniem, zdefiniuj akcję o tej samej nazwie w konfiguracji modułu. Kiedy akcja zostanie wywołana, zamiast domyślnej akcji Zabbix'a zostanie wykonana akcja modułu.

Pliki akcji powinny być przechowywane w folderze *akcje*. Akcje muszą być określone w pliku manifest.json.

Kontroler

Przepływ pracy kontrolera akcji:

1. Sprawdź, czy wszystkie parametry przekazane w żądaniu HTTP są prawidłowe:

- Wywołaj metodę *checkInput()* kontrolera;
- Użyj reguł walidacji zdefiniowanych w CNewValidator.php;
- Wywołaj metodę *validateInput()*.

2. Sprawdź uprawnienia użytkownika.
3. Przygotuj dane zgodnie z przekazanymi parametrami: jeśli *checkInput()* zwróci true, Zabbix wywoła metodę *doAction()* kontrolera.
4. Przygotuj tablicę **$data** dla widoku. Użyj *CControllerResponseData* oraz metody *setResponse()*, aby zapisać odpowiedź w tablicy **$data**.

Przykład:

```
/**
 * 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:
Pełną listę dostępnych klas kontrolerów w Zabbix można zobaczyć w kodzie źródłowym.

Widoki

Plik widoku otrzymuje dane z kontrolera, a następnie przygotowuje ich wygląd w HTML.

Note:
Definiowanie widoku(-ów) dla modułu frontend jest opcjonalne, chyba że moduł jest widgetem.
 Widgety pulpitu wymagają co najmniej dwóch widoków: jednego dla trybu edycji i jednego dla trybu podglądu (powinny być przechowywane w katalogu *views*).

W widoku można używać predefiniowanych klas HTML Zabbix (z */zabbix/ui/include/classes/html*), a także dodawać nowe klasy HTML i CSS. Nowe klasy powinny być przechowywane w folderze *assets* modułu.

Przykład:

```
...
(new CColHeader(_('Nazwa')))
```

Spowoduje to dodanie nowej nazwy kolumny *Nazwa* oraz nadanie stylu górnemu wierszowi tabeli takiego jak na innych stronach Zabbix.

Widok akcji

To jest plik referencyjny do definiowania widoku akcji.

```
<?php declare(strict_types = 1);

/**
* @var CView $this
*/

$this->includeJsFile('example.something.view.js.php');

(new CWidget())
    ->setTitle(_('Something view'))
    ->addItem(new CDiv($data['name']))
    ->addItem(new CPartial('module.example.something.reusable', [
        'contacts' => $data['contacts']
    ]))
    ->show();
```

Aktywa

Folder *assets* może zawierać dowolne pliki i podfoldery, które nie należą do innych katalogów. Możesz go używać do:

- Styli JavaScript (muszą znajdować się w *assets/js*);
- Styli CSS (muszą znajdować się w *assets/css*);
- Obrazów;
- Czcionek;
- Wszystkiego innego, co chcesz uwzględnić.

assets/js

Katalog *assets/js* jest zastrzeżony i powinien zawierać wyłącznie pliki JavaScript. Aby widget mógł używać tych plików, określ je w pliku *manifest.json*.

Na przykład:

```
"assets": {
    "js": ["class.widget.js"]
}
```

assets/css

Katalog *assets/css* jest zastrzeżony i powinien zawierać wyłącznie pliki stylu CSS. Aby widget mógł używać tych plików, określ je w pliku *manifest.json*.

Na przykład:

```
"assets": {
    "css": ["mywidget.css"]
}
```

Style CSS

Pliki CSS mogą zawierać niestandardowy atrybut `theme`, który umożliwia zdefiniowanie różnych stylów dla określonych motywów frontend.

Dostępne motywy i wartości ich atrybutów:

- **Niebieski** - [theme='blue-theme']
- **Niebieski (klasyczny)** - [theme='blue-classic-theme']
- **Ciemny** - [theme='dark-theme']
- **Ciemnoniebieski** - [theme='dark-blue-theme']
- **Ciemny (klasyczny)** - [theme='dark-classic-theme']
- **Jasny o wysokim kontraście** - [theme='hc-light']
- **Ciemny o wysokim kontraście** - [theme='hc-dark']

Przykład:

```
.widget {
    background-color: red;
}

[theme='dark-theme'] .widget {
    background-color: green;
}
```

Zarejestruj nowy moduł

Ta sekcja wyjaśnia jak dodać nowy moduł do frontendu Zabbix.

Wymagania wstępne Przed kontynuowaniem upewnij się, że:

- Moduł znajduje się w katalogu *moduły* (modules) instalacji front-endu Zabbix (na przykład *zabbix/ui/modules*).
- Moduł ma co najmniej podstawową wersję pliku manifest.json.
- Masz dostęp do sekcji menu Administracja w Zabbix (wymaga roli użytkownika Super administrator).

> **Note:**
> The frontend will not install or even recognize incompatible modules.

Dodawanie modułu Otwórz stronę *Administration* > *General* > *Modules* i kliknij *Scan directory*.



Znajdź swój moduł na liście i aktywuj go.

Aby aktywować moduł, kliknij łącze *Disabled* - stan modułu zmieni się na *Enabled*.

Kliknij nazwę modułu, aby wyświetlić dodatkowe informacje o module, takie jak autor, wersja lub krótki opis (jeśli został zdefiniowany w manifeście).

Podgląd widżetu Moduły widżetów po dodaniu stają się natychmiast widoczne na liście widżetów pulpitu.

Możesz otworzyć pulpit, przełączyć się do trybu edycji i dodać widżet do pulpitu jak zwykle.

Po wprowadzeniu zmian w widżecie odśwież pulpit, aby zobaczyć, jak widżet wygląda po najnowszych aktualizacjach.

Widżety

Widżety to moduły front-endu Zabbix używane w pulpitach nawigacyjnych. O ile nie zaznaczono inaczej, wszystkie wytyczne dotyczące modułów mają zastosowanie również do widżetów.

Jednak widżet różni się znacząco od modułu. Aby zbudować widżet:

- określ typ "widget" w pliku manifest.json ("type": "widget");
- uwzględnij co najmniej dwa widoki: jeden dla trybu prezentacji widżetu i jeden dla trybu konfiguracji widżetu (example.widget.view.php i example.widget.edit.php);
- i kontroler dla prezentacji widżetu (WidgetView.php);
- użyj i rozszerz domyślne klasy widżetów.

Konfiguracja

Konfiguracja widżetu jest częścią modułu widżetu, która pozwala użytkownikowi definiować ustawienia widżetu dla prezentacja. Ta strona opisuje klasy, które mogą być użyte do stworzenia formularza konfiguracji widżetu z niestandardowymi polami.

Widżet

Główna klasa widżetu, rozszerzająca bazową klasę wszystkich widżetów pulpitu - *CWidget*. Wymagana do nadpisania domyślnego zachowania widżetu.

Klasa Widget powinna znajdować się w katalogu głównym widżetu (na przykład, zabbix/ui/modules/my_custom_widget).

Przykład 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

Klasa *WidgetForm* rozszerza domyślną klasę *CWidgetForm* i zawiera zestaw pól *CWidgetField*, które są wymagane do zdefiniowania struktury przechowywania konfiguracji widgetu w bazie danych oraz obsługi walidacji danych wejściowych.

Klasa *WidgetForm* powinna znajdować się w katalogu *includes*. Jeśli klasa ma inną nazwę, należy ją określić w parametrze *widget/form_class* w pliku *manifest.json*.

Przykład pliku 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', _('Item')))
                    ->setFlags(CWidgetField::FLAG_NOT_EMPTY | CWidgetField::FLAG_LABEL_ASTERISK)
                    ->setMultiple(false)
            )
            ->addField(
                new CWidgetFieldTextBox('description', _('Description'))
            )
            ->addField(
                (new CWidgetFieldColor('chart_color', _('Color')))->setDefault('FF0000')
            );
    }
}
```

CWidgetFormView

Klasa *CWidgetFormView* jest wymagana do określania logiki prezentacji pól zdefiniowanych w klasie *WidgetForm* oraz ustalania ich wyglądu i zachowania podczas renderowania w widoku konfiguracji.

Klasa *CWidgetFormView* obsługuje następujące metody:

- *addField()* - przyjmuje jako parametr instancję klasy *CWidgetFieldView*; każda klasa *CWidgetField* ma odpowiednią klasę *CWidgetFieldView* do użycia w widoku konfiguracji widgetu.
- *addFieldset()* - przyjmuje instancję klasy *CWidgetFieldsGroupView*, która łączy pola w zwijany kontener.
- *addFieldsGroup()* - przyjmuje instancję klasy *CWidgetFormFieldsetCollapsibleView*, która wizualnie (za pomocą obramowania) łączy pola w grupę.
- *includeJsFile()* - umożliwia dodanie pliku JavaScript do widoku konfiguracji widgetu.
- *addJavaScript()* - umożliwia dodanie wbudowanego kodu JavaScript, który zostanie wykonany natychmiast po załadowaniu widoku konfiguracji widgetu.

Klasa *CWidgetFormView* powinna znajdować się w katalogu *views*.

Przykład views/widget.edit.php

```
<?php

/**
 * My custom widget form view.
 *
 * @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(_('Advanced configuration')))
            ->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

Klasa JavaScript może być użyta do dodania dynamicznego zachowania i interaktywności do widoku konfiguracji widżetu. Na przykład, można zainicjować selektor kolorów, zdefiniowany w klasie *CWidgetFormView*.

Klasa JavaScript powinna zostać załadowana wraz z formularzem, dlatego należy odwołać się do niej w klasie *CWidgetFormView* za pomocą metod *includeJsFile()* i *addJavaScript()*.

W poniższym przykładzie instancja klasy singleton jest natychmiast tworzona i przechowywana pod nazwą *window.my_custom_widget_form*. W ten sposób otwarcie formularza po raz drugi spowoduje ponowne utworzenie instancji.

Klasa JavaScript powinna znajdować się w katalogu *views*.

views/widget.edit.js.php example

```
<?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

Klasa *CWidgetField* jest klasą bazową, po której dziedziczą wszystkie klasy pól formularza (*CWidgetFieldCheckBox*, *CWidgetFieldTextArea*, *CWidgetFieldRadioButtonList* itd.). Klasy rozszerzające *CWidgetField* odpowiadają za odbieranie, zapisywanie i sprawdzanie poprawności wartości konfiguracji widżetu.

Dostępne są następujące klasy *CWidgetField*.

Klasa CWidgetField	Typ pola w bazie danych	Opis
CWidgetFieldCheckBox	int32	Pojedyncze pole wyboru.
CWidgetFieldCheckBoxList	array of int32	Wiele pól wyboru w ramach jednego pola konfiguracji.
CWidgetFieldColor	string	Pole wyboru koloru.
CWidgetFieldDatePicker	string	Pole wyboru daty.
CWidgetFieldIntegerBox	int32	Pole do wprowadzania liczby całkowitej. Można go użyć do skonfigurowania wartości minimalnych i maksymalnych.
CWidgetFieldLatLng	string	Pole tekstowe do wprowadzania rozdzielonych przecinkami szerokości geograficznej, długości geograficznej i poziomu powiększenia mapy.
CWidgetFieldMultiSelect	-	Klasa bazowa pól wielokrotnego wyboru, rozszerzana przez wszystkie klasy `CWidgetFieldMultiSelect*`.
CWidgetFieldMultiSelectAction	ID	Pole wielokrotnego wyboru do wybierania działań z listy działań zdefiniowanych w sekcji *Alerty* > *Działania*.
CWidgetFieldMultiSelectGraph	ID	Pole wielokrotnego wyboru do wybierania wykresów niestandardowych.
CWidgetFieldMultiSelectGraphPrototype	ID	Pole wielokrotnego wyboru do wybierania prototypów wykresów niestandardowych.
CWidgetFieldMultiSelectGroup	ID	Pole wielokrotnego wyboru do wybierania grup hostów.
CWidgetFieldMultiSelectHost	ID	Pole wielokrotnego wyboru do wybierania hostów.
CWidgetFieldMultiSelectHostInventory	array of int32	Pole wielokrotnego wyboru do wybierania pól inwentarza hosta.
CWidgetFieldMultiSelectItem	ID	Pole wielokrotnego wyboru do wybierania pozycji.
CWidgetFieldMultiSelectItemPrototype	ID	Pole wielokrotnego wyboru do wybierania prototypów pozycji.
CWidgetFieldMultiSelectMap	ID	Pole wielokrotnego wyboru do wybierania map.
CWidgetFieldMultiSelectMediaType	ID	Pole wielokrotnego wyboru do wybierania typów mediów.
CWidgetFieldMultiSelectOverrideHost	ID	Pole wielokrotnego wyboru do wybierania źródła danych dla hostów z pulpitu nawigacyjnego lub innego widżetu.
CWidgetFieldMultiSelectService	ID	Pole wielokrotnego wyboru do wybierania usług.
CWidgetFieldMultiSelectSla	ID	Pole wielokrotnego wyboru do wybierania umów SLA.
CWidgetFieldMultiSelectUser	ID	Pole wielokrotnego wyboru do wybierania użytkowników.
CWidgetFieldNavTree	array of (multiple mixed)	Przechowuje strukturę drzewa nawigacyjnego map do użycia w widżecie *Drzewo nawigacji mapy*.
CWidgetFieldNumericBox	string	Pole do wprowadzania liczby zmiennoprzecinkowej.
CWidgetFieldPatternSelect	-	Klasa bazowa pól wyboru wzorca, rozszerzana przez wszystkie klasy `CWidgetPatternSelect*`.
CWidgetFieldPatternSelectHost	string	Pole wielokrotnego wyboru do wybierania hostów. Obsługuje wzorce nazw hostów.
CWidgetFieldPatternSelectItem	string	Pole wielokrotnego wyboru do wybierania pozycji. Obsługuje wzorce nazw pozycji.
CWidgetFieldRadioButtonList	int32	Grupa przycisków radiowych składająca się z co najmniej jednego przycisku radiowego.
CWidgetFieldRangeControl	int32	Suwak do wybierania wartości typu całkowitego.
CWidgetFieldReference	string	Przechowuje unikatowy identyfikator tego widżetu na pulpicie nawigacyjnym; służy do komunikacji między widżetami.
CWidgetFieldSelect	int32	Rozwijane pole wyboru.
CWidgetFieldSeverities	array of int32	Wstępnie skonfigurowane pole *CWidgetFieldCheckBoxList* z poziomami ważności wyzwalaczy.
CWidgetFieldSparkline	array of (multiple mixed)	Umożliwia konfigurowanie wykresu typu sparkline.
CWidgetFieldTags	array of (string, int32, string)	Umożliwia konfigurowanie co najmniej jednego wiersza filtrów tagów.
CWidgetFieldTextArea	string	Pole tekstowe do wprowadzania tekstu wielowierszowego.
CWidgetFieldTextBox	string	Pole tekstowe do wprowadzania tekstu jednowierszowego.
CWidgetFieldThresholds	array of (string, string)	Umożliwia konfigurowanie par kolorów i liczb.
CWidgetFieldTimePeriod	array of string	Pole wyboru okresu.
CWidgetFieldTimeZone	string	Rozwijana lista stref czasowych.
CWidgetFieldUrl	string	Pole tekstowe do wprowadzania adresów URL.

Note:
Maksymalna długość pól `int32` wynosi 10 znaków, pól `ID` 20 znaków, a pól `string` 65535 znaków. W przypadku typów złożonych (np. `array of (string, int32, string)`) limit dotyczy każdego elementu osobno. Ograniczenia długości właściwe dla danego pola można ustawić za pomocą metod takich jak `setMaxLength()`.

Prezentacja

Ta strona opisuje komponenty, których można użyć do utworzenia widoku prezentacji widżetu. Widok prezentacji widżetu to część widżetu, która odbiera dane zgodnie z jego konfiguracją i wyświetla je na pulpicie nawigacyjnym w kontenerze.

Widok prezentacji składa się z trzech części:

- Widget action
- Widok widżetu
- JavaScript

Akcja widgetu

Klasa akcji widgetu (*WidgetView*) zawiera metody operacji na widgetach w trybie widoku prezentacji. Większość akcji widgetów używa domyślnej klasy kontrolera *CControllerDashboardWidgetView* i/lub ją rozszerza.

Klasa akcji widgetu powinna znajdować się w katalogu *actions* i być określona w parametrze *actions* (*actions/widget.{id}.view/class*) w pliku *manifest.json*.

Przykład actions/WidgetView.php (zaimplementowany w natywnym widżecie Zabbix *Informacje o systemie*)

```
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()
            ]
        ]));
    }
}
```

Widok widgetu

Klasa widoku widgetu (*CWidgetView*) odpowiada za budowanie widoku prezentacji widgetu.

Klasa widoku widgetu powinna znajdować się w katalogu *views*. Jeśli plik zawierający klasę widoku widgetu ma inną nazwę niż domyślna (*widget.view.php*), należy ją określić w parametrze *actions* pliku *manifest.json* (*actions/widget.{id}.view/view*).

Przykład views/widget.view.php

```
<?php

/**
 * Mój niestandardowy widok widgetu.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['name'])
    )
    ->show();
```

JavaScript

Klasa JavaScript odpowiada za określanie zachowania widżetu, takiego jak aktualizowanie danych widżetu, zmiana rozmiaru widżetu, wyświetlanie elementów widżetu itp.

Wszystkie operacje JavaScript używają i/lub rozszerzają bazową klasę JavaScript wszystkich widżetów pulpitu - *CWidget*. Klasa *CWidget* zawiera zestaw metod z domyślną implementacją zachowania widżetu. W zależności od złożoności widżetu metody te mogą być używane bez zmian lub rozszerzane.

Klasa *CWidget* zawiera następujące metody:

- Metody definiujące cykl życia widżetu: *onInitialize()*, *onStart()*, *onActivate()*, *onDeactivate()*, *onDestroy()*, *onEdit()*.
- Metody obsługujące aktualizację i wyświetlanie danych widżetu: *promiseUpdate()*, *getUpdateRequestData()*, *processUpdateResponse(response)*, *processUpdateErrorResponse(error)*, *setContents(response)*.
- Metody modyfikujące wygląd widżetu: *onResize()*, *hasPadding()*.

Klasa JavaScript powinna znajdować się w katalogu *assets/js* i być określona w parametrze *assets* (*assets/js*) w pliku *manifest.json*.

Metody cyklu życia

Metody cyklu życia widżetu są wywoływane przez pulpit nawigacyjny oraz na różnych etapach cyklu życia widżetu podczas jego istnienia w pulpicie nawigacyjnym.

Metoda ***onInitialize()*** definiuje początkowy stan i/lub wartości widżetu, bez wykonywania jakichkolwiek manipulacji HTML lub danymi. Metoda ta jest wywoływana, gdy tworzony jest widżet (tworzona jest instancja obiektu widżetu), zazwyczaj poprzez dodanie widżetu do strony pulpitu nawigacyjnego lub załadowanie strony pulpitu nawigacyjnego.

Przykład:

```
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;
}
```

Metoda ***onStart()*** definiuje strukturę HTML widgetu, bez wykonywania jakiejkolwiek manipulacji danymi. Ta metoda jest wywoływana przed pierwszą aktywacją strony pulpitu, czyli zanim pulpit i jego widgety zostaną w pełni wyświetlone użytkownikowi.

Przykład:

```
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`
        );
    }
}
```

Metoda ***onActivate()*** uaktywnia widget i czyni go interaktywnym poprzez włączenie niestandardowych detektorów zdarzeń (służących do reagowania na działania użytkownika) oraz zainicjowanie cyklu aktualizacji widgetu (aby utrzymywać jego zawartość na bieżąco). Ta metoda jest wywoływana po aktywacji strony pulpitu, czyli gdy zostanie ona w pełni wyświetlona w interfejsie użytkownika.

Należy pamiętać, że przed wywołaniem metody *onActivate()* widget znajduje się w stanie nieaktywnym (`WIDGET_STATE_INACTIVE`). Po pomyślnym wywołaniu widget przechodzi do stanu aktywnego (`WIDGET_STATE_ACTIVE`). W stanie aktywnym widget reaguje na działania, nasłuchuje zdarzeń, okresowo aktualizuje swoją zawartość i może współdziałać z innymi widgetami.

Przykład:

```
onActivate() {
    this._startClock();

    this._resize_observer = new ResizeObserver(this._events.resize);
    this._resize_observer.observe(this._target);
}
```

Metoda ***onDeactivate()*** zatrzymuje wszelką aktywność i interaktywność widżetu przez dezaktywację niestandardowych nasłuchiwaczy zdarzeń oraz zatrzymanie cyklu aktualizacji widżetu. Ta metoda jest wywoływana, gdy strona pulpitu jest dezaktywowana,

to znaczy po przełączeniu na inną stronę lub usunięciu, albo gdy widżet zostanie usunięty ze strony pulpitu.

Należy pamiętać, że przed wywołaniem metody *onDeactivate()* widżet znajduje się w stanie aktywnym (`WIDGET_STATE_ACTIVE`). Po pomyślnym wywołaniu widżet przechodzi do stanu nieaktywnego (`WIDGET_STATE_INACTIVE`).

Przykład:

```
onDeactivate() {
    this._stopClock();
    this._resize_observer.disconnect();
}
```

Metoda ***onDestroy()*** wykonuje zadania porządkowe przed usunięciem widgetu z dashboardu, co może obejmować zamknięcie połączenia z bazą danych ustanowionego podczas inicjalizacji widgetu, usunięcie danych tymczasowych w celu zwolnienia pamięci systemowej i uniknięcia wycieków zasobów, wyrejestrowanie detektorów zdarzeń związanych ze zdarzeniami zmiany rozmiaru lub kliknięciami przycisków, aby zapobiec niepotrzebnej obsłudze zdarzeń i wyciekom pamięci itd. Ta metoda jest wywoływana, gdy widget lub strona dashboardu, która go zawiera, zostaje usunięta.

Należy pamiętać, że przed wywołaniem metody *onDestroy()* widget w stanie aktywnym (`WIDGET_STATE_ACTIVE`) jest zawsze dezaktywowany przez wywołanie metody *onDeactivate()*.

Przykład:

```
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);
    }
}
```

Metoda ***onEdit()*** definiuje wygląd i zachowanie widżetu, gdy pulpit przechodzi w tryb edycji. Ta metoda jest wywoływana, gdy pulpit przechodzi w tryb edycji, zazwyczaj gdy użytkownik użyje przycisku *Edytuj* widżetu lub przycisku *Edytuj pulpit* na pulpicie.

Przykład:

```
onEdit() {
    this._deactivateGraph();
}
```

Metoda procesu aktualizacji

Metoda procesu aktualizacji widżetu są odpowiedzialne za pobieranie zaktualizowanych danych z serwera Zabbix lub dowolnego innego źródła danych i wyświetlanie ich w widżecie.

Metoda ***promiseUpdate()*** inicjuje proces aktualizacji danych przez pobieranie danych, zazwyczaj za pomocą żądań web lub wywołań API. Ta metoda jest wywoływana, gdy strona pulpitu jest wyświetlana, a następnie okresowo, aż strona pulpitu zostanie przełączona na inną stronę pulpitu.

Poniżej przedstawiono przykład domyślnej implementacji metody *promiseUpdate()* używanej przez większość natywnych widgetów Zabbix. W domyślnej implementacji metoda *promiseUpdate()* stosuje ogólny wzorzec pobierania danych z serwera. Tworzy nowy obiekt `Curl` z odpowiednim adresem URL i parametrami żądania, wysyła żądanie `POST` za pomocą metody *fetch()* z obiektem danych zbudowanym przez metodę *getUpdateRequestData()*, a następnie odpowiednio przetwarza odpowiedź (lub odpowiedź błędu) za pomocą *processUpdateResponse(response)* lub *processUpdateErrorResponse(error)*. Ta implementacja jest odpowiednia dla większości widgetów, ponieważ zazwyczaj pobierają one dane w formacie JSON i obsługują je w spójny sposób.

```
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);
        });
    }
```

Metoda ***getUpdateRequestData()*** przygotowuje dane żądania do serwera na potrzeby aktualizacji widżetu, zbierając różne właściwości i odpowiadające im wartości (identyfikatory widżetu, ustawienia filtrów, zakresy czasu itp.) ze stanu i konfiguracji widżetu, a następnie tworząc obiekt danych reprezentujący informacje wymagane do wysłania do serwera w żądaniu aktualizacji. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *promiseUpdate()*, czyli podczas procesu aktualizacji widżetu.

Domyślna implementacja:

```
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
    };
}
```

Metoda ***processUpdateResponse(response)*** obsługuje odpowiedź otrzymaną z serwera po żądaniu aktualizacji i, jeśli proces aktualizacji zakończył się powodzeniem i bez błędów, czyści dane widżetu oraz wyświetla nową zawartość za pomocą metody *setContents()*. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *promiseUpdate()*, czyli podczas procesu aktualizacji widżetu.

Domyślna implementacja:

```
processUpdateResponse(response) {
    this._setHeaderName(response.name);

    this._updateMessages(response.messages);
    this._updateInfo(response.info);
    this._updateDebug(response.debug);

    this.setContents(response);
}
```

Metoda ***processUpdateErrorResponse(error)*** obsługuje odpowiedź otrzymaną z serwera po żądaniu aktualizacji, jeśli odpowiedź zawiera błąd, i wyświetla komunikat/-y o błędzie. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *promiseUpdate()*, czyli podczas procesu aktualizacji widżetu.

Domyślna implementacja:

```
processUpdateErrorResponse(error) {
    this._updateMessages(error.messages, error.title);
}
```

Metoda ***setContents(response)*** wyświetla zawartość widżetu, jeśli proces aktualizacji widżetu zakończył się pomyślnie i bez błędów, co może obejmować manipulowanie elementami DOM, aktualizowanie komponentów interfejsu użytkownika, stosowanie stylów lub formatowania itp. Ta metoda jest wywoływana wyłącznie jako część domyślnej metody *processUpdateResponse(response)*, czyli podczas obsługi odpowiedzi otrzymanej z serwera po żądaniu aktualizacji.

Domyślna implementacja:

```
setContents(response) {
    this._body.innerHTML = response.body ?? '';
}
```

Metoda modyfikacji prezentacji

Metoda modyfikacji prezentacji widżetu jest odpowiedzialna za modyfikację wyglądu widżetu.

Metoda ***onResize()*** odpowiada za dostosowanie elementów wizualnych widgetu do nowego rozmiaru widgetu, co może obejmować zmianę rozmieszczenia elementów, dostosowanie wymiarów elementów, obcinanie tekstu, implementację lazy loading w celu poprawy responsywności podczas zmiany rozmiaru itp. Ta metoda jest wywoływana, gdy widget zmienia rozmiar, na przykład gdy użytkownik ręcznie zmienia rozmiar widgetu lub gdy zmieniany jest rozmiar okna przeglądarki.

Przykład:

```
onResize() {
    if (this.getState() === WIDGET_STATE_ACTIVE) {
        this._startUpdating();
    }
}
```

Metoda ***hasPadding()*** odpowiada za zastosowanie pionowego dopełnienia 8 px u dołu widżetu, gdy jest on skonfigurowany tak, aby wyświetlać nagłówek. Ta metoda jest wywoływana po aktywowaniu strony pulpitu, czyli gdy staje się ona stroną wyświetlaną w interfejsie użytkownika.

Implementacja domyślna:

```
hasPadding() {
    return this.getViewMode() !== ZBX_WIDGET_VIEW_MODE_HIDDEN_HEADER;
}
```

W przypadku niektórych widżetów konieczne jest wykorzystanie całej dostępnej przestrzeni widżetu do skonfigurowania na przykład niestandardowego koloru tła. Poniżej przedstawiono przykład implementacji metody *hasPadding()* używanej w natywnym widżecie Zabbix *Wartość pozycji*.

```
hasPadding() {
    return false;
}
```

Poradniki

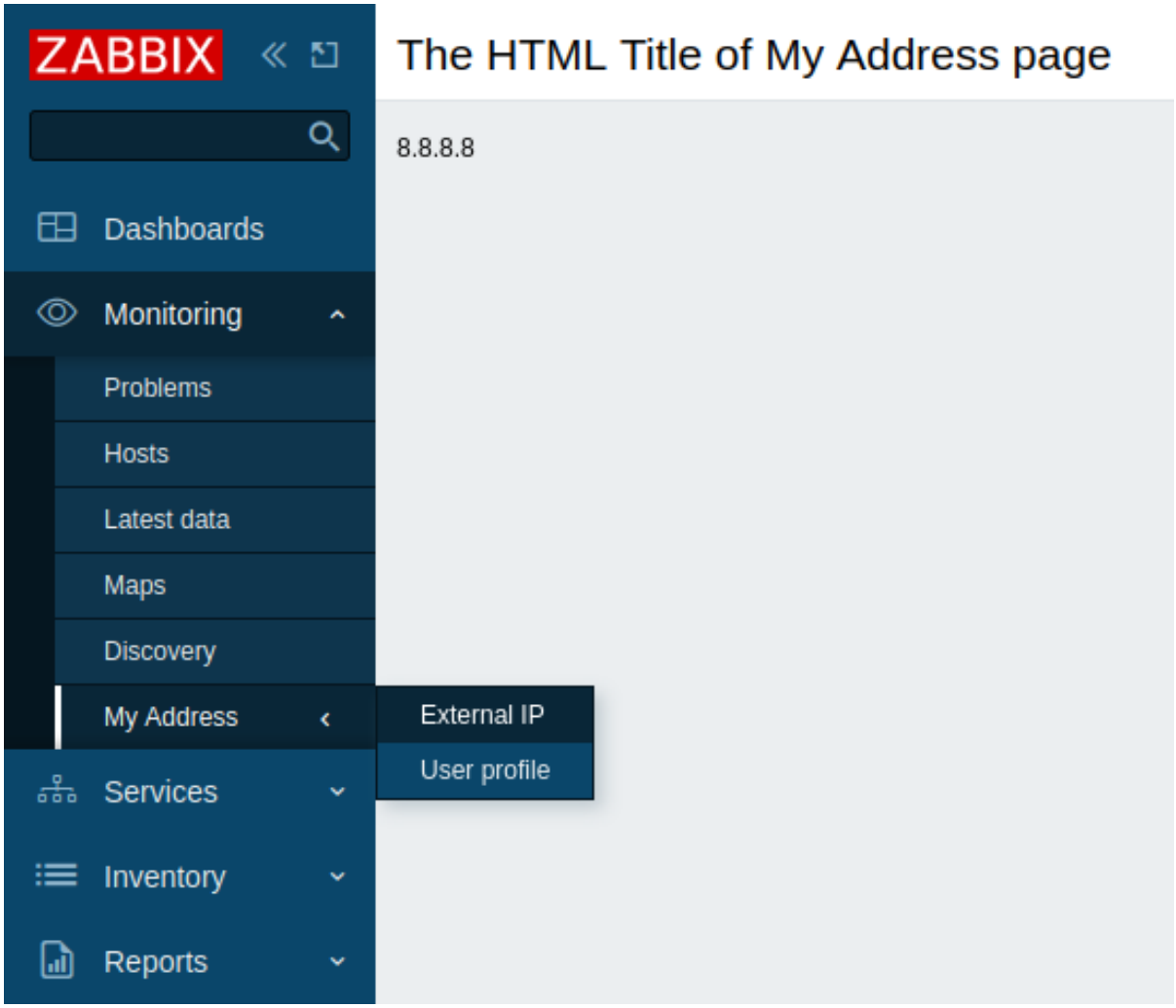
Ta sekcja zawiera praktyczne instrukcje krok po kroku, które pokazują, jak stworzyć własny moduł oraz widżet w Zabbix.

Utwórz moduł (samouczek)

To jest samouczek krok po kroku, który pokazuje, jak utworzyć prosty moduł front-endu Zabbix. Możesz pobrać wszystkie pliki tego modułu jako archiwum ZIP: MyAddress.zip.

Co zbudujesz

Podczas tego samouczka najpierw zbudujesz moduł frontend, który doda nową sekcję menu *My Address*, a następnie przekształcisz go w bardziej zaawansowany moduł frontend, który wykonuje żądanie HTTP do *https://api.seeip.org* i wyświetla odpowiedź — adres IP twojego komputera — na nowej stronie w nowo utworzonej sekcji menu *My Address*. Oto jak będzie wyglądał gotowy moduł:



Część I - Podstawowy moduł

Dodaj pusty moduł do frontend Zabbix

1. Utwórz katalog *MyAddress* w katalogu *modules* instalacji frontend Zabbix (na przykład *zabbix/ui/modules*).
2. Utwórz plik *manifest.json* z podstawowymi metadanymi modułu (zobacz opis obsługiwanych parametrów).

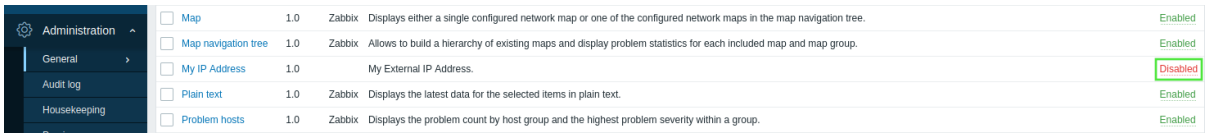
ui/modules/MyAddress/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "my-address",
    "name": "My IP Address",
    "version": "1.0",
    "namespace": "MyAddress",
    "description": "My External IP Address."
}
```

3. W frontend Zabbix przejdź do sekcji *Administration* > *General* > *Modules* i kliknij przycisk *Scan directory*.



4. Znajdź nowy moduł *My IP Address* na liście i kliknij hiperłącze ”Disabled”, aby zmienić status modułu z ”Disabled” na ”Enabled” (jeśli moduł nie jest widoczny na liście, zobacz sekcję troubleshooting).



Moduł jest teraz zarejestrowany w frontend. Nie jest jednak jeszcze widoczny, ponieważ nadal trzeba zdefiniować jego funkcjonalność. Po dodaniu zawartości do katalogu modułu zmiany będą natychmiast widoczne w frontend Zabbix po odświeżeniu strony.

Utwórz sekcję menu

1. Utwórz plik *Module.php* w katalogu *MyAddress*.

Ten plik implementuje nową klasę *Module*, która rozszerza domyślną klasę *CModule*. Klasa *Module* doda nową sekcję menu *My Address* do głównego menu.

Metoda *setAction()* określa akcję, która ma zostać wykonana po kliknięciu sekcji menu. Na początek możesz użyć predefiniowanej akcji *userprofile.edit*, która otworzy stronę *User profile*. W części III tego samouczka dowiesz się, jak utworzyć własną akcję.

ui/modules/MyAddress/Module.php

```
<?php

namespace Modules\MyAddress;

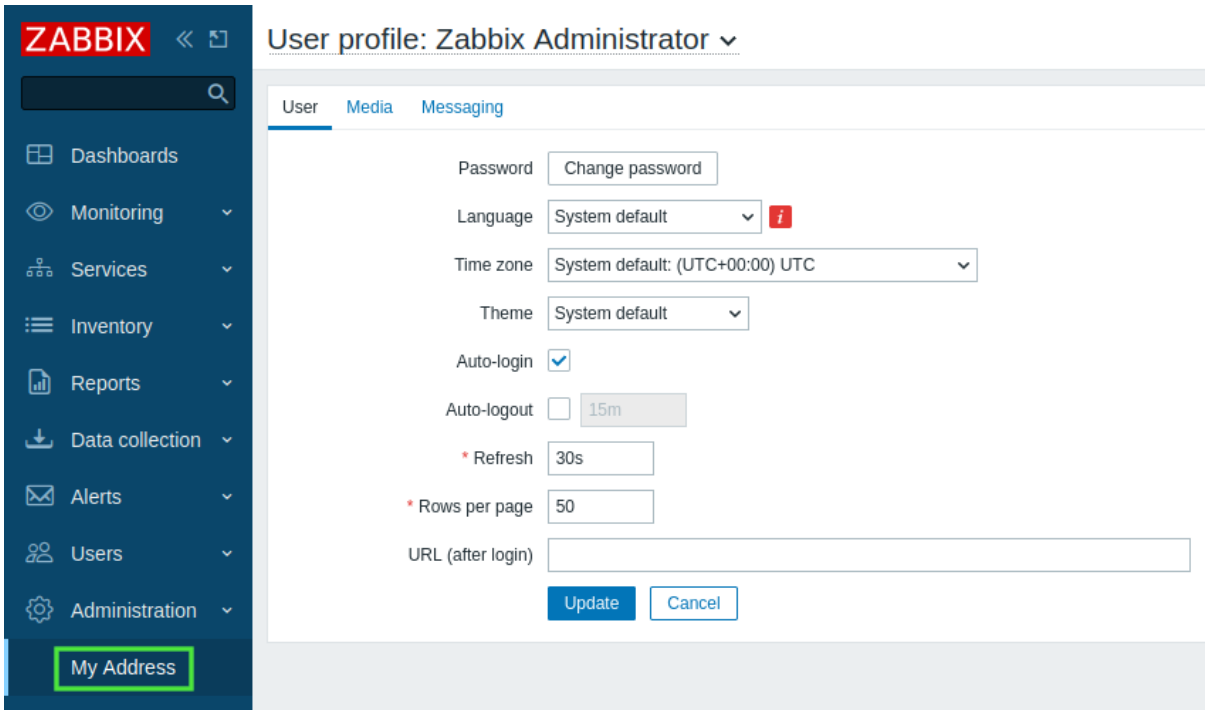
use Zabbix\Core\CModule,
    APP,
    CMenuItem;

class Module extends CModule {

    public function init(): void {
        APP::Component()->get('menu.main')
            ->add((new CMenuItem(_('My Address')))
            ->setAction('userprofile.edit'));
    }
}
```

Note:
Możesz zastąpić `'userprofile.edit'` innymi akcjami, na przykład `'charts.view'` (otwiera niestandardowe wykresy), `'problems.view'` (otwiera *Monitoring* > *Problems*) lub `'report.status'` (otwiera raport *System information*).

3. Odśwież frontend Zabbixa. Na dole głównego menu Zabbixa pojawiła się teraz nowa sekcja *My Address*. Kliknij *My Address*, aby otworzyć stronę *User profile*.



Część II - zmiana lokalizacji sekcji menu

W tej części przeniesiesz sekcję menu *My Address* do sekcji *Monitoring*, a następnie dodasz do niej zagnieżdżone menu. W rezultacie użytkownicy będą mogli uzyskać dostęp do dwóch podstron z sekcji menu *Monitoring* > *My Address*.

1. Otwórz i edytuj plik *Module.php*.

ui/modules/MyAddress/Module.php

```php
<?php

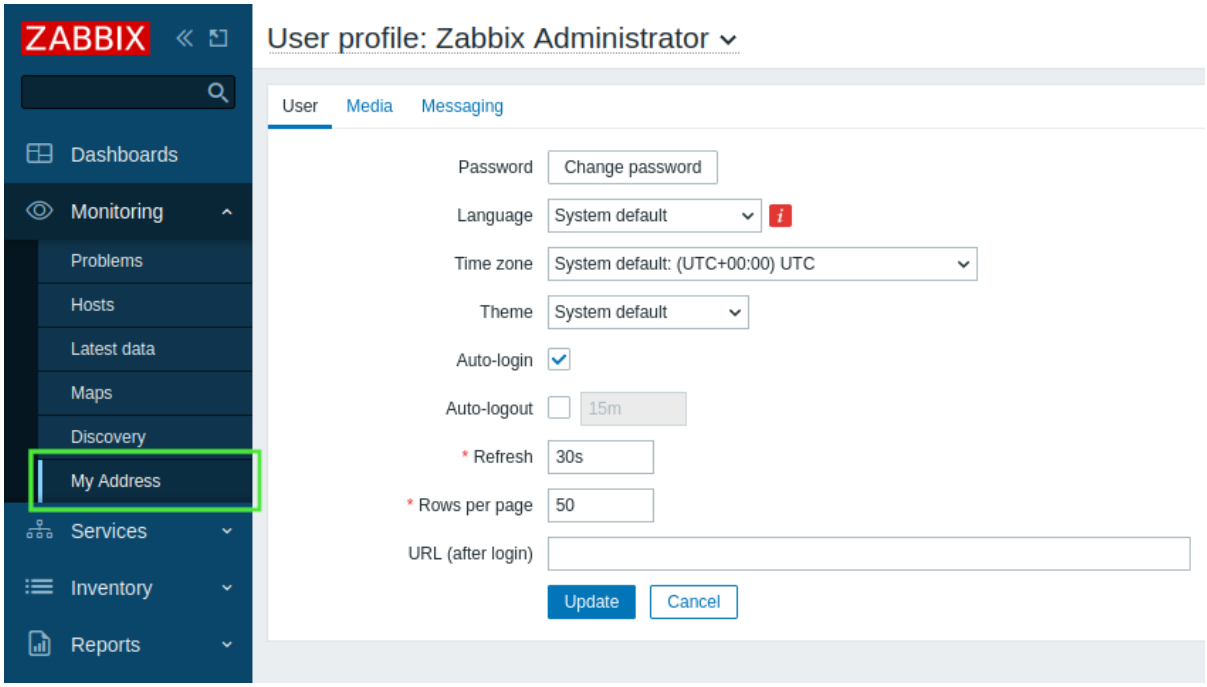
namespace Modules\MyAddress;

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. Odśwież frontend Zabbixa. Rozwiń sekcję menu *Monitoring* i sprawdź, że sekcja *My address* znajduje się teraz poniżej sekcji *Discovery*.



3. Aby dodać zagnieżdżone strony do sekcji menu *My Address*, ponownie otwórz i edytuj plik *Module.php*.

Ten krok tworzy dwie podsekcje:

- *External IP*, która wykonuje nową akcję 'my.address', zdefiniowaną w kolejnych krokach;
- *User profile*, która wykonuje predefiniowaną akcję 'userprofile.edit', aby otworzyć stronę *User profile*.

Zwróć uwagę, że dla zagnieżdżonego menu musisz użyć klasy *CMenu* oprócz klas użytych w poprzednich krokach.

ui/modules/MyAddress/Module.php

```php
<?php

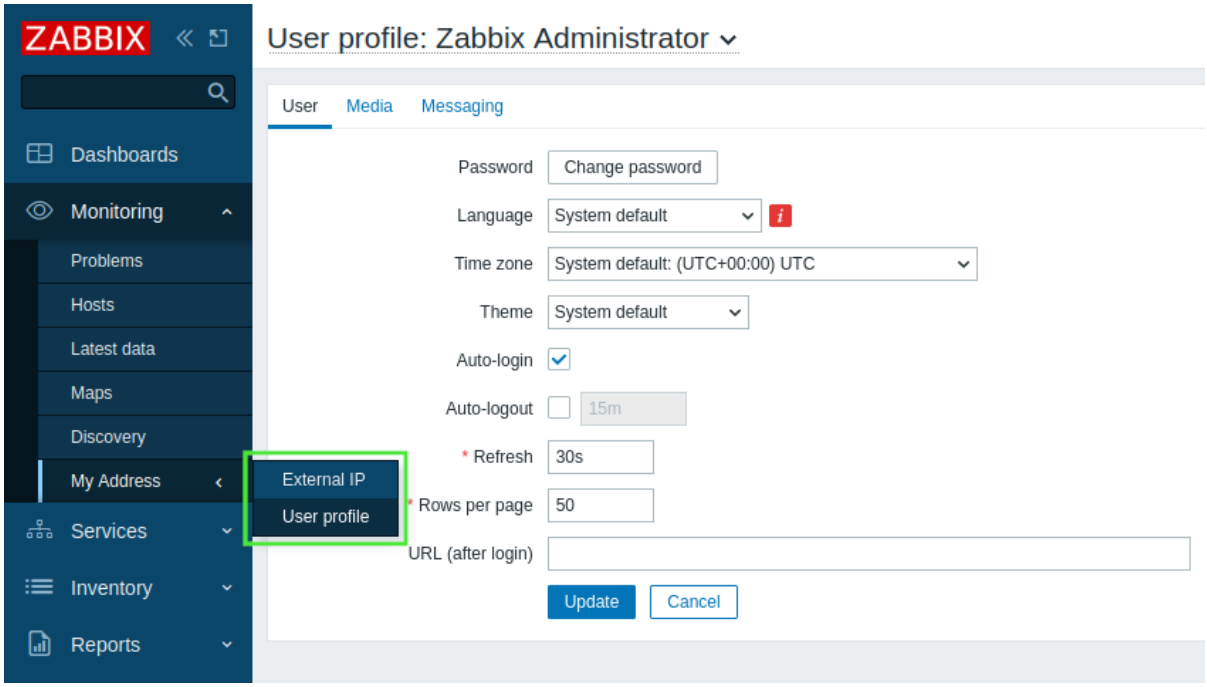
namespace Modules\MyAddress;

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. Odśwież frontend Zabbixa. Zauważ, że sekcja menu *My address* zawiera teraz menu trzeciego poziomu z dwiema stronami - *External IP* i *User profile*.



Część III - Akcja modułu

Akcja jest zaimplementowana w dwóch plikach - *actions/MyAddress.php* i *views/my.address.php*. Plik ***actions/MyAddress.php*** odpowiada za implementację logiki biznesowej, natomiast plik ***views/my.address.php*** odpowiada za widok.

1. Utwórz katalog *actions* w katalogu *MyAddress*.
2. Utwórz plik *MyAddress.php* w katalogu *actions*.

Logika akcji zostanie zdefiniowana w klasie *MyAddress*. Ta klasa akcji zaimplementuje cztery funkcje: *init()*, *checkInput()*, *checkPermissions()* oraz *doAction()*. frontend Zabbixa wywołuje funkcję *doAction()* wtedy, gdy akcja zostanie zażądana. Ta funkcja odpowiada za logikę biznesową modułu.

Attention:
Dane muszą być zorganizowane jako tablica asocjacyjna. Tablica może być wielowymiarowa i może zawierać dowolne dane oczekiwane przez widok.

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. Utwórz nowy katalog *views* w katalogu *MyAddress*.
4. Utwórz plik *my.address.php* w katalogu *views* i zdefiniuj widok modułu.

Zwróć uwagę, że zmienna `$data` jest dostępna w widoku bez konieczności jej specjalnego definiowania. Framework automatycznie przekazuje tablicę asocjacyjną do widoku.

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();
```

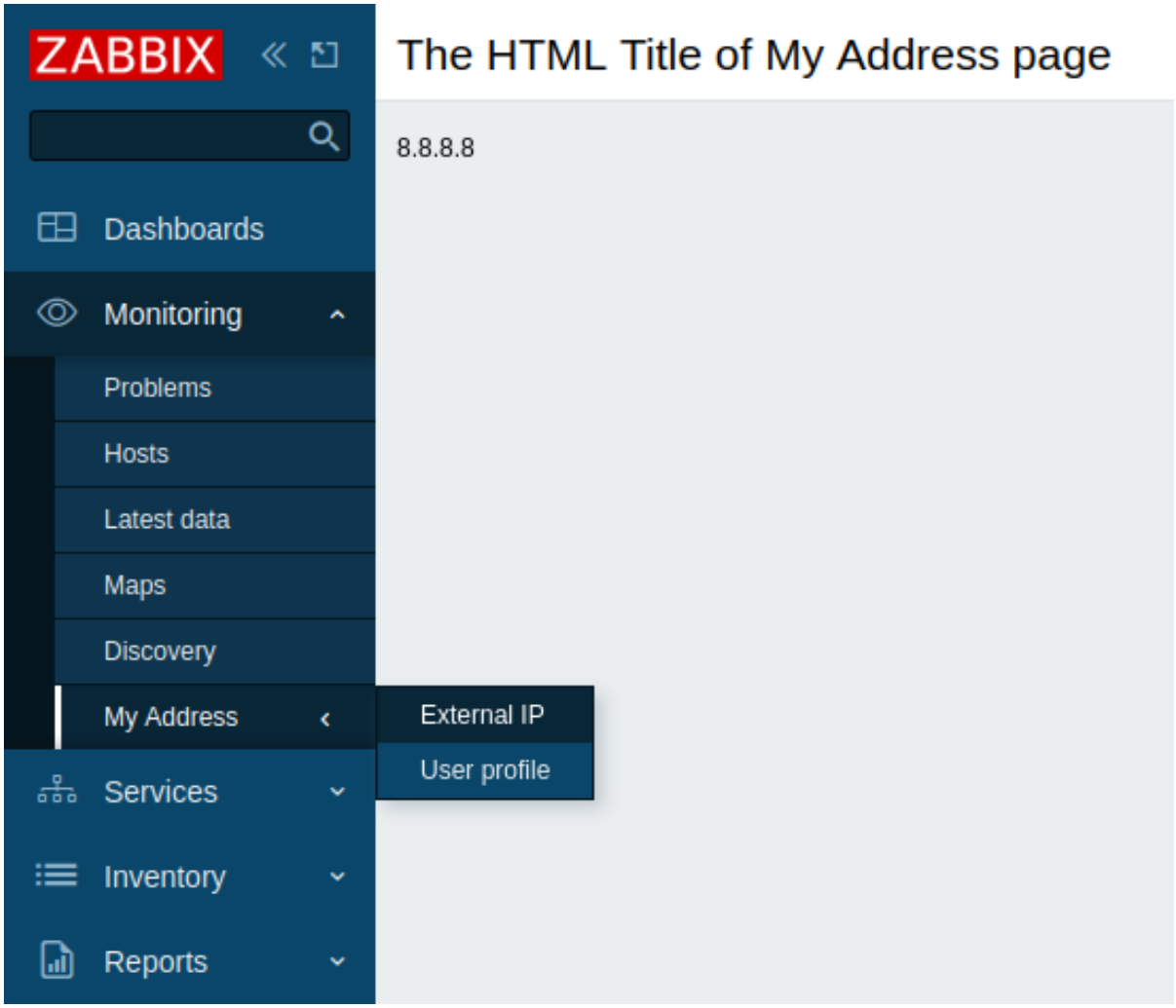
5. Akcja modułu musi zostać zarejestrowana w pliku *manifest.json*. Otwórz *manifest.json* i dodaj nowy obiekt `actions`, który zawiera:

- klucz akcji z nazwą akcji zapisaną małymi literami (a-z) i ze słowami oddzielonymi kropkami (na przykład `my.address`);
- nazwę klasy akcji (`MyAddress`) jako wartość klucza `class` obiektu `my.address`;
- nazwę widoku akcji (`my.address`) jako wartość klucza `view` obiektu `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"
        }
    }
}
```

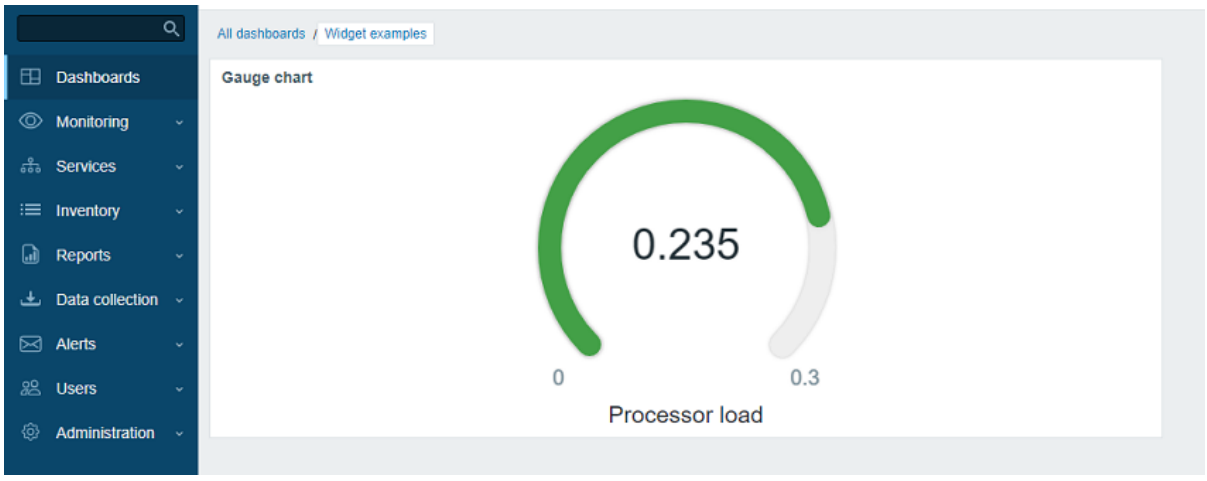
6. Odśwież frontend Zabbixa. Kliknij *My address* > *External IP*, aby zobaczyć adres IP swojego komputera.



Utwórz widżet (samouczek)

To jest samouczek pokazujący krok po kroku, jak utworzyć prosty widget pulpitu nawigacyjnego. Możesz pobrać wszystkie pliki tego widgetu w postaci archiwum ZIP: lesson_gauge_chart.zip.

Co zbudujesz Podczas tego samouczka najpierw utworzysz podstawowy widget „Hello, world!”, a następnie przekształcisz go w bardziej zaawansowany widget, który wyświetla wartość pozycji jako wykres wskaźnikowy. Tak będzie wyglądać gotowy widget:



Część I - ”Witaj, świecie!” W tej sekcji dowiesz się, jak utworzyć minimalną wymaganą liczbę elementów widgetu i dodać nowy widget do frontendu Zabbix.

Dodawanie pustego widżetu do frontend Zabbix

1. Utwórz katalog `lesson_gauge_chart` w katalogu `modules` instalacji frontend Zabbix (na przykład *zabbix/ui/modules*).

Note:
Wszystkie niestandardowe widżety są traktowane jako moduły zewnętrzne i muszą zostać dodane do katalogu `modules` instalacji frontend Zabbix (na przykład *zabbix/ui/modules*). Katalog `zabbix/ui/widgets` jest zarezerwowany dla wbudowanych widżetów Zabbix i jest aktualizowany wraz z frontend Zabbix.

2. Utwórz plik `manifest.json` z podstawowymi metadanymi widżetu (zobacz opis obsługiwanych parametrów).

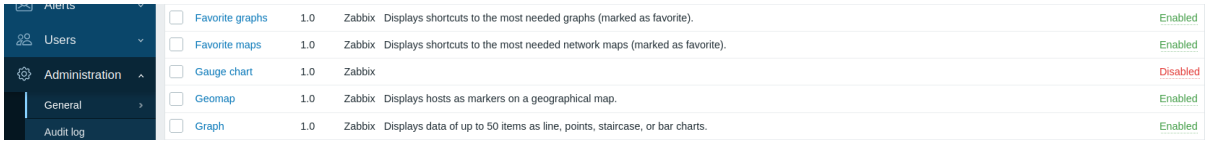
ui/modules/lesson_gauge_chart/manifest.json

```
{
    "manifest_version": 2.0,
    "id": "lesson_gauge_chart",
    "type": "widget",
    "name": "Gauge chart",
    "namespace": "LessonGaugeChart",
    "version": "1.1",
    "author": "Zabbix"
}
```

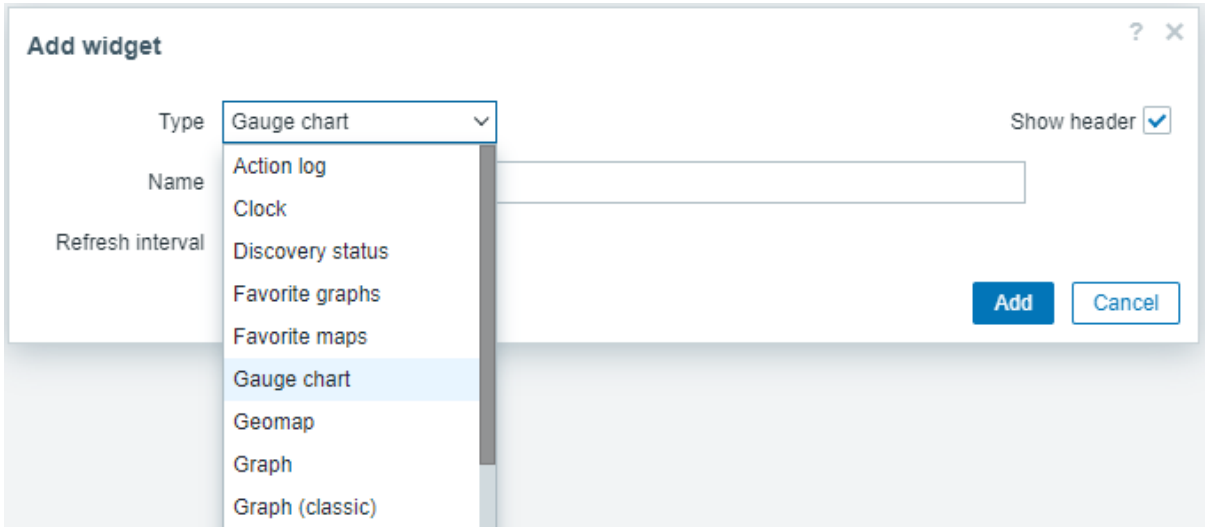
3. W frontend Zabbix przejdź do sekcji *Administration* > *General* > *Modules* i kliknij przycisk *Scan directory*.



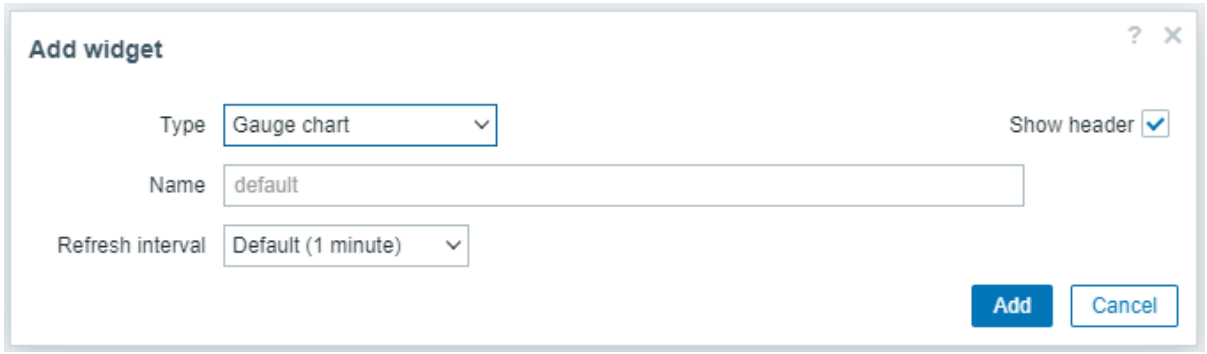
4. Znajdź nowy moduł *Gauge chart* na liście i kliknij hiperłącze "Disabled", aby zmienić status modułu z "Disabled" na "Enabled" (jeśli moduł nie jest widoczny na liście, zobacz sekcję troubleshooting).



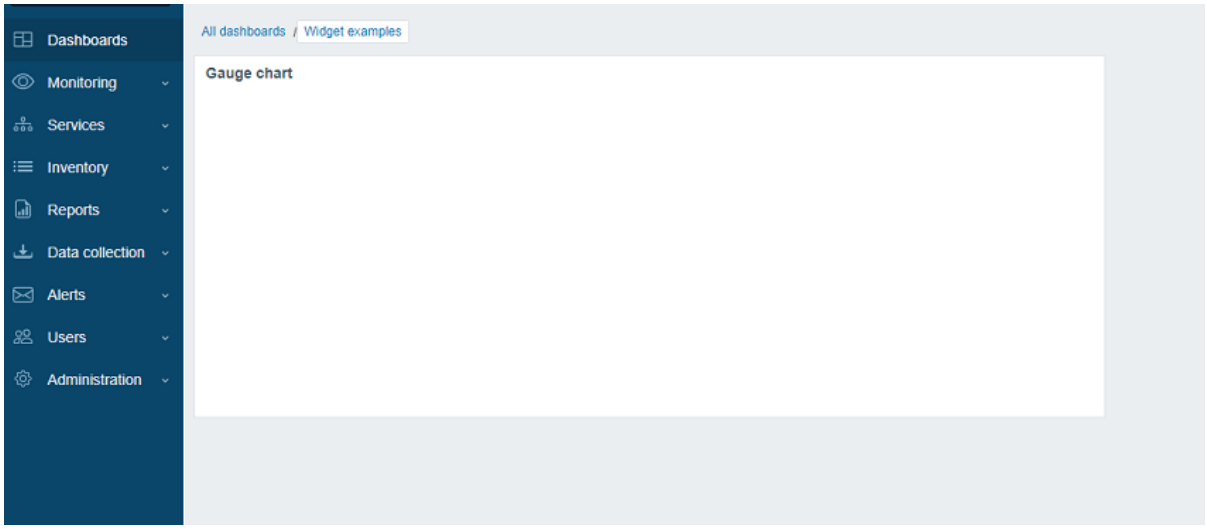
5. Otwórz dashboard, przełącz go w tryb edycji i dodaj nowy widżet. W polu *Type* wybierz "Gauge chart".



6. W tym momencie konfiguracja widżetu *Gauge chart* zawiera tylko ogólne pola widżetu *Name* i *Refresh interval*. Kliknij *Add*, aby dodać widżet do dashboardu.



7. Na dashboardzie powinien pojawić się pusty widżet. Kliknij *Save changes* w prawym górnym rogu, aby zapisać dashboard.



Dodaj widok widgetu

Note:
Plik **widoku** widgetu powinien znajdować się w katalogu `views` (w tym samouczku: `ui/modules/lesson_gauge_chart/views/`). Jeśli plik ma domyślną nazwę `widget.view.php`, nie trzeba go rejestrować w pliku `manifest.json`. Jeśli plik ma inną nazwę, określ go w sekcji `actions/widget.lesson_gauge_chart.view` pliku `manifest.json`.

1. Utwórz katalog `views` w katalogu `lesson_gauge_chart`.
2. Utwórz plik `widget.view.php` w katalogu `views`.

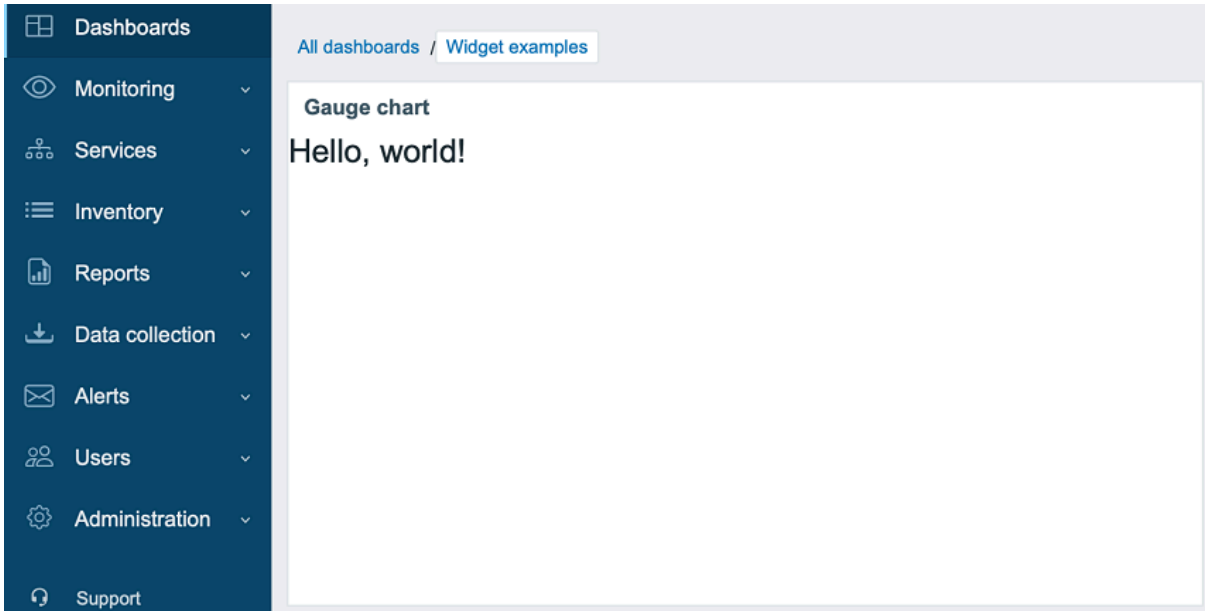
`ui/modules/lesson_gauge_chart/views/widget.view.php`

```
<?php

/**
 * Widok widgetu wykresu Gauge.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, 'Hello, world!')
    )
    ->show();
```

3. Odśwież pulpit. Widget *Gauge chart* wyświetla teraz tekst "Hello, world!".



Część II - Wykres wskaźnikowy Dodawanie ustawień do widoku konfiguracji i używanie ich w widoku widżetu

W tej sekcji dowiesz się, jak dodać pole konfiguracji widżetu i wyświetlić wprowadzoną wartość w widoku widżetu jako tekst.

Konfiguracja widżetu składa się z formularza (`Zabbix\Widgets\CWidgetForm`) oraz widoku formularza widżetu (`widget.edit.php`). Aby dodać pola (`Zabbix\Widgets\CWidgetField`), musisz utworzyć klasę `WidgetForm`, która będzie dziedziczyć po `Zabbix\Widgets\CWidgetForm`.

Formularz zawiera zestaw pól (`Zabbix\Widgets\CWidgetField`) różnych typów, które służą do walidacji wartości wprowadzonych przez użytkownika. Pole formularza (`Zabbix\Widgets\CWidgetField`) dla każdego typu elementu wejściowego konwertuje wartość do pojedynczego formatu, aby zapisać ją w bazie danych.

> **Note:**
> Plik **formularza** widżetu powinien znajdować się w katalogu `includes` (na potrzeby tego samouczka: `ui/modules/lesson_gauge_chart/includes/`). Jeśli plik ma domyślną nazwę `WidgetForm.php`, nie musisz rejestrować go w pliku `manifest.json`. Jeśli plik ma inną nazwę, określ ją w sekcji `widget/form_class` pliku `manifest.json`.

1. Utwórz nowy katalog `includes` w katalogu `lesson_gauge_chart`.
2. Utwórz plik `WidgetForm.php` w katalogu `includes`.

`ui/modules/lesson_gauge_chart/includes/WidgetForm.php`

```
<?php

namespace Modules\LessonGaugeChart\Includes;

use Zabbix\Widgets\CWidgetForm;

class WidgetForm extends CWidgetForm {
}
```

3. Dodaj pole *Description* do formularza konfiguracji widżetu. Jest to zwykłe pole tekstowe, w którym użytkownik może wprowadzić dowolny zestaw znaków. Możesz użyć do tego klasy `CWidgetFieldTextBox`.

`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. W katalogu `views` utwórz plik widoku konfiguracji widżetu `widget.edit.php` i dodaj widok dla nowego pola *Description*. Dla klasy pola `CWidgetFieldTextBox` widokiem jest `CWidgetFieldTextBoxView`.

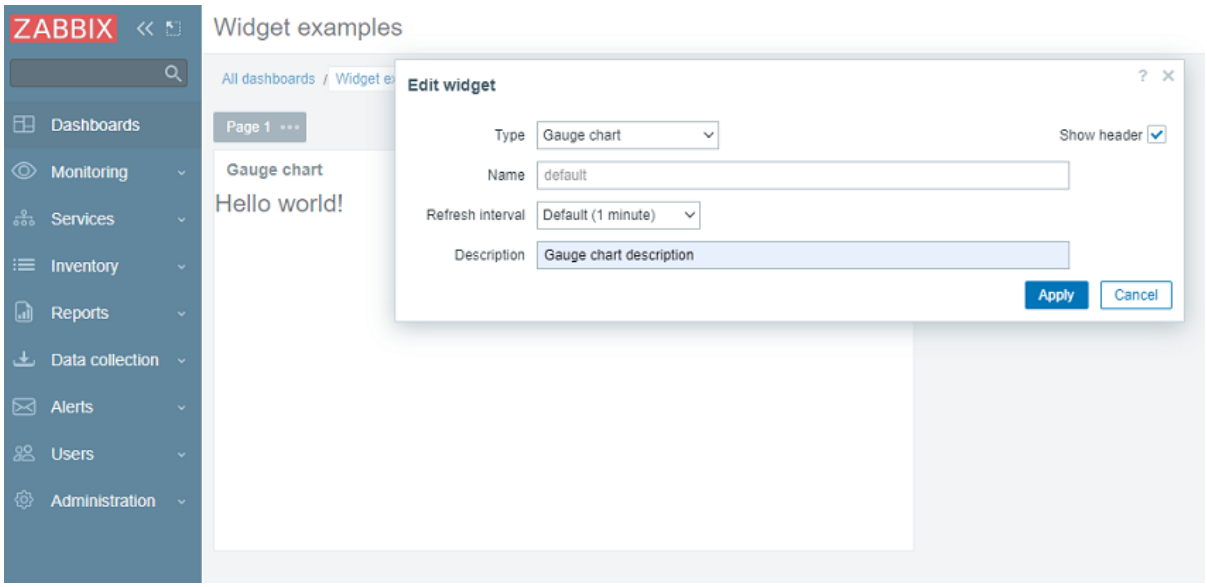
`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```
<?php

/**
 * Gauge chart widget form view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetFormView($data))
    ->addField(
        new CWidgetFieldTextBoxView($data['fields']['description'])
    )
    ->show();
```

5. Przejdź do dashboardu i kliknij ikonę koła zębatego w widżecie, aby otworzyć formularz konfiguracji widżetu.
6. Formularz konfiguracji widżetu zawiera teraz nowe pole tekstowe *Description*. Wpisz dowolną wartość, na przykład *Gauge chart description*.



7. Kliknij *Apply* w formularzu konfiguracji widżetu. Następnie kliknij *Save changes* w prawym górnym rogu, aby zapisać dashboard. Zwróć uwagę, że nowy opis nie jest widoczny nigdzie, a widżet nadal wyświetla "Hello, world!".

Aby nowy opis pojawił się w widżecie, wartość pola *Description* musi zostać pobrana z bazy danych i przekazana do widoku widżetu. W tym celu musisz utworzyć klasę akcji.

8. Utwórz nowy katalog `actions` w katalogu `lesson_gauge_chart`.
9. Utwórz plik `WidgetView.php` w katalogu `actions`. Klasa akcji `WidgetView` będzie dziedziczyć po klasie `CControllerDashboardWi`

Wartości pól konfiguracji widżetu są przechowywane we właściwości `$fields_values` klasy akcji.

`ui/modules/lesson_gauge_chart/actions/WidgetView.php`

```
<?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. Otwórz `manifest.json` i zarejestruj `WidgetView` jako klasę akcji w sekcji `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. Teraz możesz użyć wartości pola opisu, zawartej w `$data['description']`, w widoku widżetu. Otwórz `views/widget.view.php` i zastąp statyczny tekst "Hello, world!" wartością `$data['description']`.

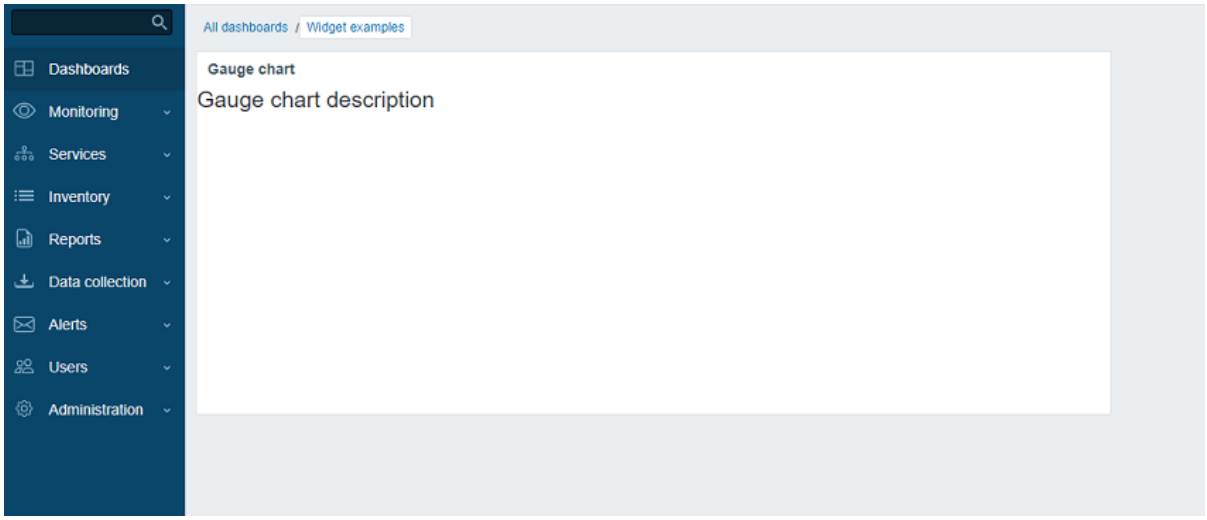
ui/modules/lesson_gauge_chart/views/widget.view.php

```php
<?php

/**
 * Gauge chart widget view.
 *
 * @var CView $this
 * @var array $data
 */

(new CWidgetView($data))
    ->addItem(
        new CTag('h1', true, $data['description'])
    )
    ->show();
```

12. Odśwież stronę dashboardu. Powinieneś teraz zobaczyć tekst opisu widżetu zamiast "Hello, world!".



Pobieranie wartości pozycji przez API

Widget powinien wyświetlać ostatnią wartość pozycji wybranej przez użytkownika. W tym celu należy dodać możliwość wyboru pozycji w konfiguracji widgetu.

W tej sekcji dowiesz się, jak dodać pole wyboru pozycji do formularza widgetu oraz jak dodać wizualną część tego pola do widoku konfiguracji. Następnie kontroler widgetu będzie mógł pobierać dane pozycji i jej wartość za pomocą żądania API. Po otrzymaniu wartości można ją wyświetlić w widoku widgetu.

1. Otwórz `includes/WidgetForm.php` i dodaj pole `CWidgetFieldMultiSelectItem`. Umożliwi to wybór pozycji w formularzu konfiguracji.

`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. Otwórz `views/widget.edit.php` i dodaj wizualny komponent pola do widoku konfiguracji.

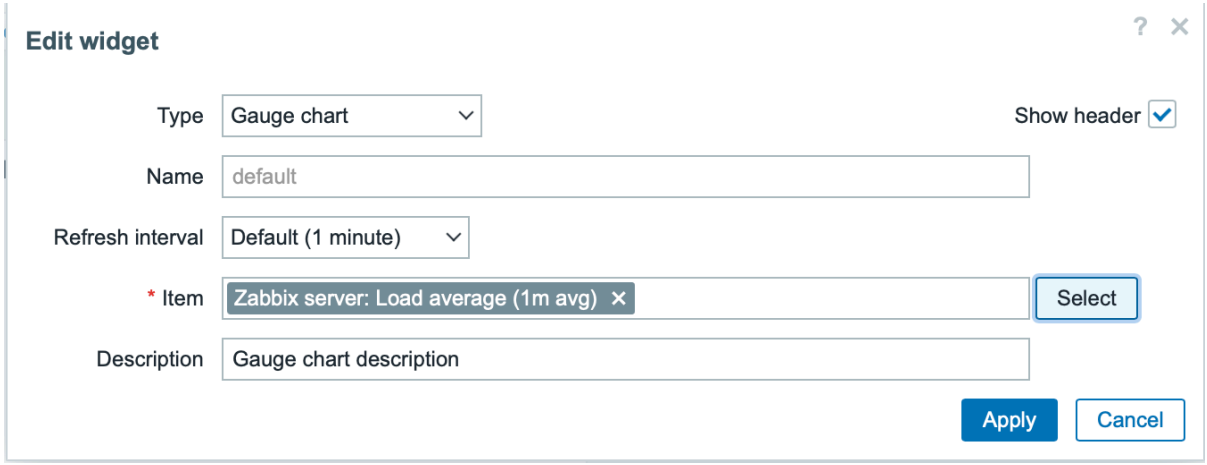
`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```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. Wróć do dashboardu i kliknij ikonę koła zębatego w widgetcie, aby otworzyć formularz konfiguracji widgetu.
4. Formularz konfiguracji widgetu zawiera teraz nowe pole wejściowe *Item*. Wybierz hosta "Zabbix server" oraz pozycję "Load average (1m avg)".



5. Kliknij *Apply* w formularzu konfiguracji widgetu. Następnie kliknij *Save changes* w prawym górnym rogu, aby zapisać dashboard.
6. Otwórz i zmodyfikuj `actions/WidgetView.php`.

Od tego momentu identyfikator pozycji będzie dostępny w kontrolerze widgetu w `$this->fields_values['itemid']`. Metoda kontrolera `doAction()` pobiera dane pozycji (nazwę, typ wartości, jednostki) za pomocą metody API `item.get` oraz ostatnią wartość pozycji za pomocą metody 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. Otwórz `views/widget.view.php` i dodaj wartość pozycji do widoku widgetu.

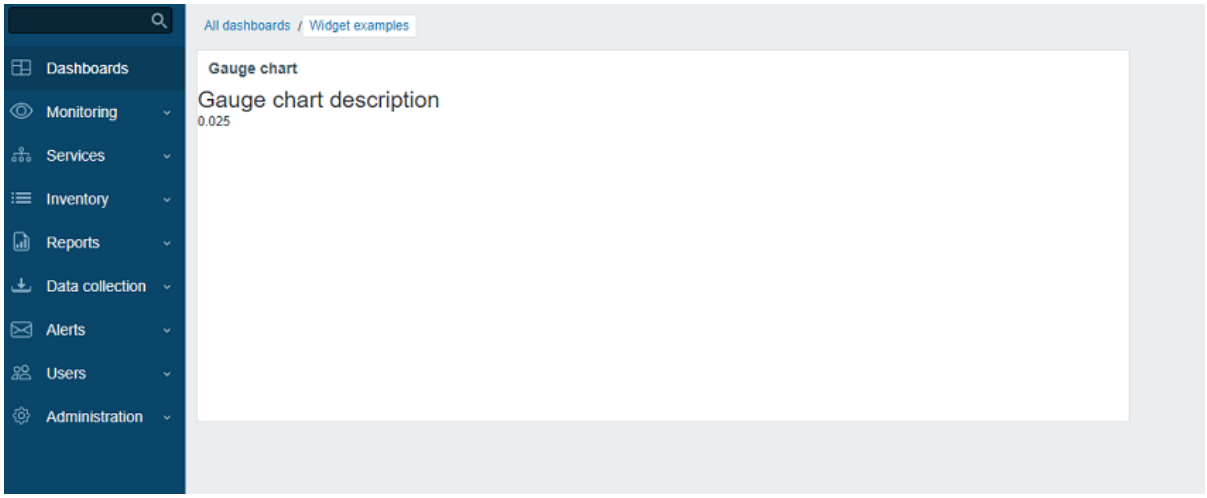
`ui/modules/lesson_gauge_chart/views/widget.view.php`

```
<?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. Odśwież stronę dashboardu. Widget wyświetli najnowszą wartość pozycji.



Dodawanie zaawansowanych ustawień konfiguracji do widoku konfiguracji

W tej sekcji dowiesz się, jak dodać rozwijaną/zwijaną sekcję *Advanced configuration* z opcjonalnymi parametrami, takimi jak kolor, wartości minimalna i maksymalna, jednostki oraz pole *Description* utworzone wcześniej.

1. Utwórz plik `Widget.php` w głównym katalogu widgetu `lesson_gauge_chart`, aby utworzyć nową klasę `Widget`.

Klasa `Widget` będzie dziedziczyć po klasie bazowej `CWidget`, aby dodać/nadpisać domyślne ustawienia widgetu (w tym przypadku - tłumaczenia). Poniższy kod JavaScript wyświetla ciąg "No data" w przypadku braku danych. Ciąg "No data" znajduje się w plikach tłumaczeń interfejsu Zabbix.

Jeśli istnieją jakiekolwiek stałe widgetu, zaleca się również określić je w klasie `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. Otwórz `includes/WidgetForm.php` i dodaj nowe pola *Color* (selektor koloru), *Min* (pole numeryczne), *Max* (pole numeryczne) oraz *Units* (lista wyboru), a także zdefiniuj domyślną paletę kolorów dla selektora koloru, aby można było z niej skorzystać w kolejnych krokach.

`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. Otwórz `views/widget.edit.php` i dodaj wizualne komponenty pól do widoku konfiguracji.

`ui/modules/lesson_gauge_chart/views/widget.edit.php`

```
<?php

/**
 * Gauge chart widget form view.
 *
 * @var CView $this
```

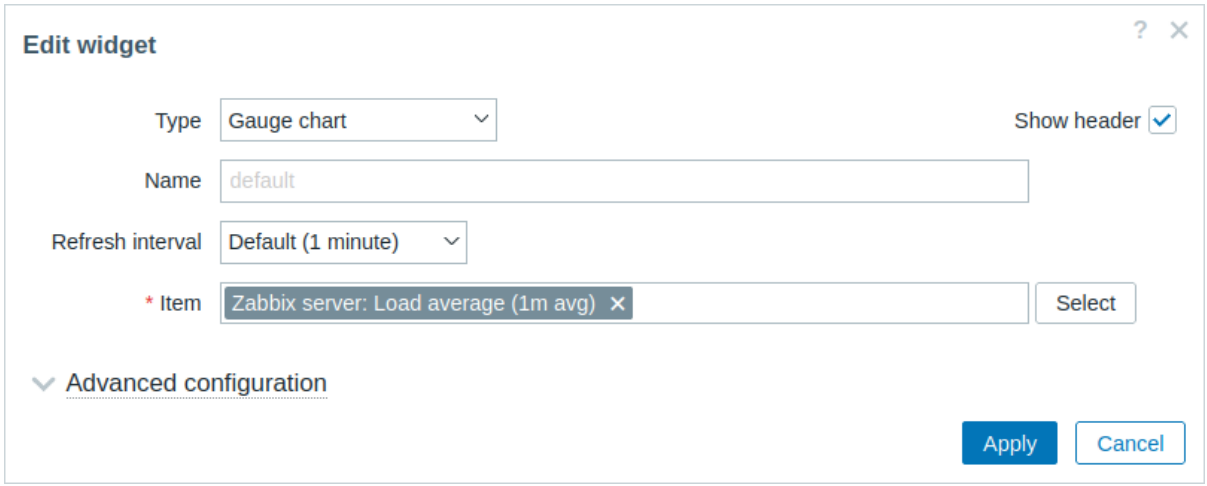
```
 * @var array $data
 */

$lefty_units = new CWidgetFieldSelectView($data['fields']['value_units']);
$lefty_static_units = (new CWidgetFieldTextBoxView($data['fields']['value_static_units']))
    ->setPlaceholder(_('value'))
    ->setWidth(ZBX_TEXTAREA_TINY_WIDTH);

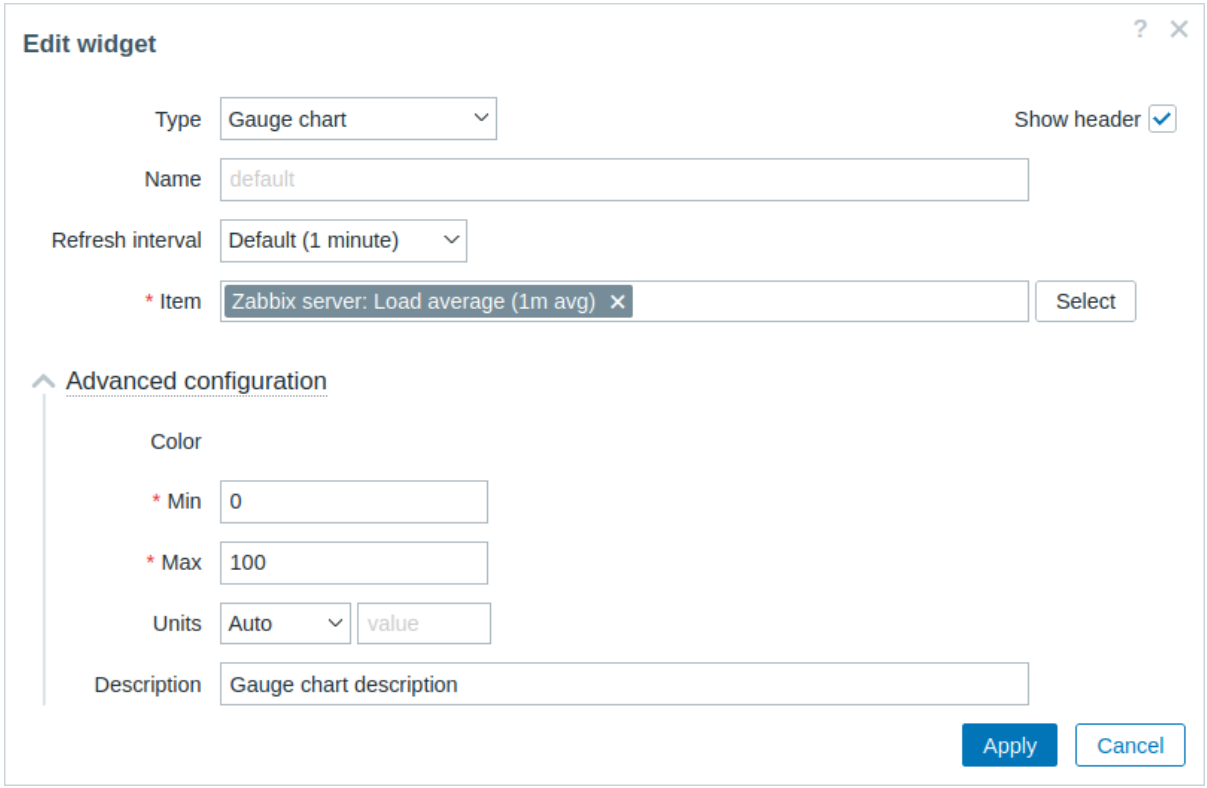
(new CWidgetFormView($data))
    ->addField(
        (new CWidgetFieldMultiSelectItemView($data['fields']['itemid']))
            ->setPopupParameter('numeric', true)
    )
    ->addFieldset(
        (new CWidgetFormFieldsetCollapsibleView(_('Advanced configuration')))
            ->addField(
                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:
Metoda `addField()` klasy `CWidgetFormView` przyjmuje ciąg klasy CSS jako drugi parametr.

4. Wróć do dashboardu, przełącz się w tryb edycji i kliknij ikonę koła zębatego w widgetcie, aby otworzyć formularz konfiguracji widgetu. Formularz konfiguracji widgetu zawiera teraz nową rozwijaną/zwijaną sekcję *Advanced configuration*.



5. Rozwiń sekcję *Advanced configuration*, aby zobaczyć dodatkowe pola konfiguracji widgetu. Zwróć uwagę, że pole *Color* nie ma jeszcze selektora koloru. Dzieje się tak, ponieważ selektor koloru musi zostać zainicjalizowany za pomocą JavaScript, który zostanie dodany w następnej sekcji - *Add JavaScript to the widget*.



Dodawanie JavaScript do widgetu

W tej sekcji dowiesz się, jak dodać wykres wskaźnikowy - utworzony przy użyciu JavaScript - który pokazuje, czy najnowsza wartość jest prawidłowa, czy też zbyt wysoka/zbyt niska.

1. Utwórz plik `widget.edit.js.php` w katalogu `views`.

JavaScript będzie odpowiedzialny za inicjalizację selektora kolorów w widoku konfiguracji.

`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. Otwórz `views/widget.edit.php` i dodaj plik `widget.edit.js.php` z JavaScript do widoku konfiguracji. Aby to zrobić, użyj metody `includeJsFile()`. Aby dodać JavaScript wbudowany, użyj metody `addJavaScript()`.

ui/modules/lesson_gauge_chart/views/widget.edit.php

```
<?php

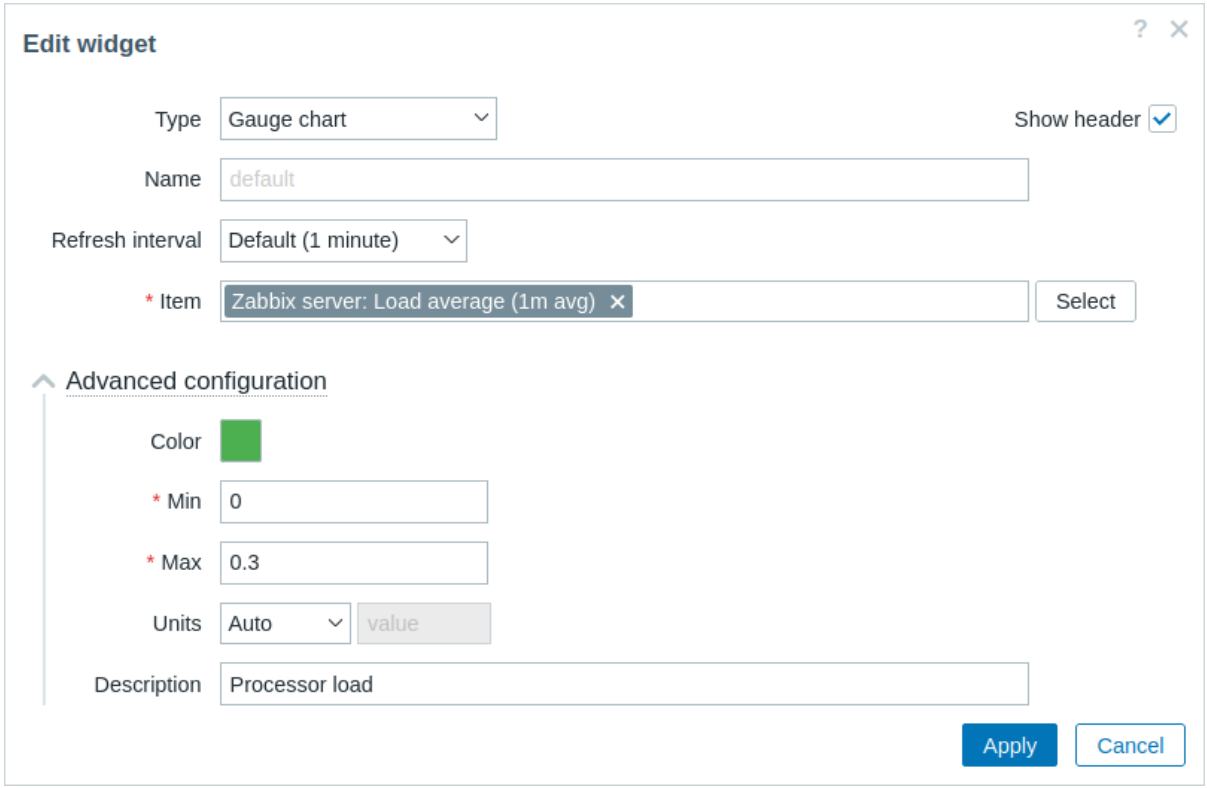
/**
 * Widok formularza widgetu wykresu wskaźnikowego.
 *
 * @var CView $this
 * @var array $data
 */

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. Wróć do dashboardu, kliknij ikonę koła zębatego w widgetcie, aby otworzyć formularz konfiguracji widgetu. Następnie rozwiń sekcję *Advanced configuration*, aby zobaczyć zainicjalizowany selektor kolorów. Wypełnij pola wartościami i wybierz kolor dla wykresu wskaźnikowego.



4. Kliknij *Apply* w formularzu konfiguracji widgetu. Następnie kliknij *Save changes* w prawym górnym rogu, aby zapisać dashboard.

5. Otwórz `actions/WidgetView.php` i zaktualizuj kontroler.

Właściwość `$this->fields_values` zawiera teraz wartości wszystkich pól *Advanced configuration*. Dokończ kontroler, aby umożliwić przekazanie konfiguracji i wybranej wartości elementu do widoku widgetu.

`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. Otwórz i zmodyfikuj `views/widget.view.php`.

Musisz utworzyć kontener dla wykresu wskaźnikowego, który narysujesz w kolejnych krokach, oraz kontener dla opisu.

Aby przekazać wartości do JavaScript jako obiekt JSON, użyj metody `setVar()`.

`ui/modules/lesson_gauge_chart/views/widget.view.php`

```
<?php

/**
 * Widok widgetu wykresu wskaźnikowego.
 *
 * @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. Utwórz nowy katalog `assets` w katalogu `lesson_gauge_chart`. Ten katalog będzie używany do przechowywania JavaScript, CSS oraz potencjalnie innych zasobów, takich jak czcionki lub obrazy.
8. Dla JavaScript widoku widgetu utwórz katalog `js` w katalogu `assets`.
9. Utwórz plik `class.widget.js` w katalogu `assets/js`.

Ta klasa JavaScript widgetu będzie rozszerzać podstawową klasę JavaScript wszystkich widgetów dashboardu - `CWidget`.

Dashboard opiera się na poprawnej implementacji widgetu i przekazuje wszelkie istotne informacje do widgetu poprzez wywoływanie odpowiednich metod JavaScript. Dashboard oczekuje również, że widget będzie generował zdarzenia, gdy wystąpi jakaś interakcja. Dlatego klasa `CWidget` zawiera zestaw metod z domyślną implementacją zachowania widgetu, którą można dostosować przez rozszerzenie klasy.

W tym przypadku potrzebne jest pewne dostosowanie, dlatego niestandardowa logika zostanie zaimplementowana dla następujących zachowań widgetu:

- inicjalizacja widgetu, odpowiedzialna za zdefiniowanie początkowego stanu widgetu (zobacz metodę `onInitialize()`);
- wyświetlanie zawartości widgetu (czyli rysowanie wykresu wskaźnikowego), jeśli proces aktualizacji widgetu zakończył się pomyślnie i bez błędów (zobacz metodę `processUpdateResponse(response)` oraz powiązane metody `_resizeChart()` i `_updatedChart()`)
- zmiana rozmiaru widgetu (zobacz metodę `onResize()` oraz powiązaną metodę `_resizeChart()`)

Dla pozostałych aspektów widgetu wykresu wskaźnikowego zostanie użyta domyślna implementacja zachowania widgetu. Aby dowiedzieć się więcej o metodach JavaScript klasy `CWidget`, zobacz: JavaScript.

Ponieważ ten JavaScript jest wymagany dla widoku widgetu, powinien być ładowany wraz ze stroną dashboardu. Aby włączyć ładowanie JavaScript, należy zaktualizować parametry `assets/js` i `js_class` w pliku `manifest.json`, jak pokazano w kroku 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. Otwórz `manifest.json` i dodaj:

- nazwę pliku (`class.widget.js`) do tablicy w sekcji `assets/js`;
- nazwę klasy (`WidgetLessonGaugeChart`) do parametru `js_class` w sekcji `widget`.

Klasa `WidgetLessonGaugeChart` będzie teraz automatycznie ładowana wraz z dashboardem.

`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"]
    }
}
```

Dodawanie stylów CSS do widgetu

W tej sekcji dowiesz się, jak dodać niestandardowe style CSS, aby widget wyglądał bardziej atrakcyjnie.

1. Aby dodać style widgetu, utwórz nowy katalog `css` w katalogu `assets`.
2. Utwórz plik `widget.css` w katalogu `assets/css`. Aby stylizować elementy widgetu, użyj selektora `div.dashboard-widget-{widget id}`. Aby skonfigurować CSS dla całego widgetu, użyj selektora `form.dashboard-widget-{widget 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;
}
```

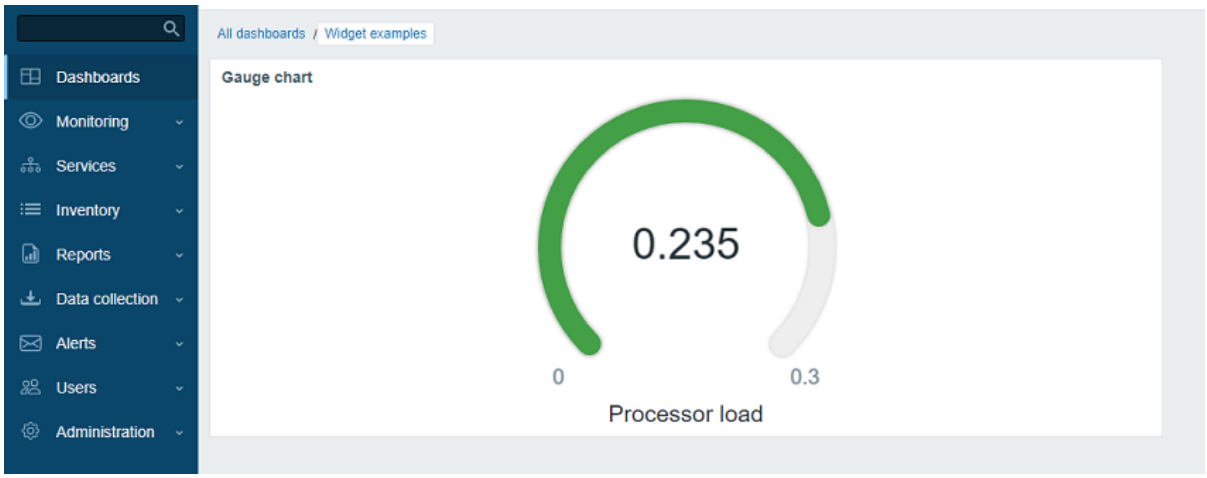
3. Otwórz plik `manifest.json` i dodaj nazwę pliku CSS (`widget.css`) do tablicy w sekcji `assets/css`. Umożliwi to wczytywanie stylów CSS zdefiniowanych w `widget.css` wraz ze stroną dashboardu.

`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. Odśwież stronę dashboardu, aby zobaczyć gotową wersję widgetu.



Przykłady

Ta sekcja zawiera pliki przykładowych modułów i widgetów, których możesz użyć jako podstawy do tworzenia własnych modułów.

Aby użyć modułu:

1. Pobierz archiwum ZIP.
2. Rozpakuj zawartość do oddzielnego katalogu w katalogu *modules* Twojej instalacji frontowej Zabbix (na przykład *zabbix/ui/modules*).
3. Zarejestruj moduł w interfejsie frontu Zabbix.

Przykład modułu Podczas tworzenia grupy hostów, przyznaj uprawnienia do odczytu skonfigurowanym grupom użytkowników - hg_auto_perm.zip

Przykłady widgetów

- Minimalny widget - widget_min.zip
- Widget ”Hello, world” wykorzystujący wyłącznie CSS - hello_world_css.zip
- Widget ”Hello, world” wykorzystujący wyłącznie JavaScript - hello_world_js.zip
- Widget ”Hello, world” wykorzystujący PHP - hello_world_php.zip

Jako przykładów możesz także użyć natywnych widgetów Zabbix’a.

Zmiany w tworzeniu rozszerzeń

Zmiany z 7.4 do 8.0

Strony podręcznika Zabbix

Dokumentacja referencyjna narzędzi i usług Zabbix wiersza poleceń, zawierająca listę dostępnych poleceń, opcji i przykładów użycia.

zabbix_agent2

```
ZABBIX_AGENT2(8)            Podręcznik administratora systemu           ZABBIX_AGENT2(8)

NAZWA
       zabbix_agent2 - Zabbix agent 2

SKŁADNIA
       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

OPIS
       zabbix_agent2  jest  aplikacją do monitorowania parametrów różnych
       usług.

OPCJE
       -c, --config config-file
              Użyj alternatywnego pliku config-file zamiast domyślnego.

       -R, --runtime-control runtime-option
              Wykonaj funkcje administracyjne zgodnie z runtime-option.

           Opcje kontroli w czasie działania
               userparameter_reload
                   Przeładuj parametry użytkownika z pliku konfiguracyjnego

               log_level_increase
                   Zwiększ poziom logowania

               log_level_decrease
                   Zmniejsz poziom logowania

               help
                   Wyświetl dostępne opcje kontroli w czasie działania

               metrics
                   Wyświetl dostępne metryki
               version
                   Wyświetl wersję

       -p, --print
              Wyświetl znane pozycje i zakończ działanie. Dla każdej pozycji
              używane są albo ogólne wartości domyślne, albo dostarczane są
              określone wartości domyślne do testowania. Te wartości domyślne
              są wymienione w nawiasach kwadratowych jako parametry klucza
              pozycji. Zwracane wartości są ujęte w nawiasy kwadratowe i po-
              przedzone typem zwracanej wartości, oddzielonym znakiem piono-
              wej kreski. Dla parametrów użytkownika typ jest zawsze t, po-
              nieważ agent nie może określić wszystkich możliwych wartości
              zwracanych. Pozycje wyświetlane jako działające nie mają gwa-
              rancji działania z serwera Zabbix ani przez zabbix_get podczas
```

```
              odpytywania działającego demona agent, ponieważ uprawnienia lub
              środowisko mogą być inne. Typy zwracanych wartości to:

              d   Liczba z częścią dziesiętną.

              m   Nieobsługiwane. Może to być spowodowane odpytywaniem pozycji,
                  która działa tylko w trybie aktywnym, jak pozycja monitorowa-
                  nia logu, albo pozycji wymagającej wielu zebranych wartości.
                  Problemy z uprawnieniami lub nieprawidłowe parametry użytkow-
                  nika również mogą skutkować stanem nieobsługiwanym.

              s   Tekst. Maksymalna długość nie jest ograniczona.

              t   Tekst. To samo co s.

              u   Nieujemna liczba całkowita.

       -t, --test item-key
              Przetestuj pojedynczą pozycję i zakończ działanie. Opis danych
              wyjściowych znajduje się w --print.

       -v, --verbose
              Włącz szczegółowe dane wyjściowe podczas testowania pozycji lub
              wyświetlania znanych pozycji.

       -T, --test-config
              Zweryfikuj plik konfiguracyjny i zakończ działanie.

       -h, --help
              Wyświetl tę pomoc i zakończ działanie.

       -V, --version
              Wyświetl informacje o wersji i zakończ działanie.

PLIKI
       /usr/local/etc/zabbix_agent2.conf
              Domyślna lokalizacja pliku konfiguracyjnego Zabbix agent 2 (jeśli
              nie została zmodyfikowana podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       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)            Podręcznik administratora systemu           ZABBIX_AGENTD(8)

NAZWA
       zabbix_agentd - demon agenta Zabbix

SKŁADNIA
       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

OPIS
       zabbix_agentd jest demonem do monitorowania różnych parametrów serwera.

OPCJE
       -c, --config config-file
              Użyj alternatywnego pliku config-file zamiast domyślnego.

       -f, --foreground
              Uruchom agenta Zabbix na pierwszym planie.

       -R, --runtime-control runtime-option
              Wykonaj funkcje administracyjne zgodnie z runtime-option.

           Opcje kontroli w czasie działania
               userparameter_reload
                   Przeładuj parametry użytkownika z pliku konfiguracyjnego

               log_level_increase[=target]
                   Zwiększ poziom logowania; wpływa na wszystkie procesy, jeśli
                   target nie został określony

               log_level_decrease[=target]
                   Zmniejsz poziom logowania; wpływa na wszystkie procesy, jeśli
                   target nie został określony

           Cele kontroli poziomu logowania
               process-type
                   Wszystkie procesy określonego typu (active checks, collector,
                   listener)

               process-type,N
                   Typ procesu i numer (np. listener,3)
               pid Identyfikator procesu, do 65535. Dla większych wartości określ
                   target jako "process-type,N"

       -p, --print
              Wyświetl znane pozycje i zakończ. Dla każdej pozycji używane są
              albo ogólne wartości domyślne, albo dostarczane są specyficzne
              wartości domyślne do testowania. Te wartości domyślne są wyświetlane
              w nawiasach kwadratowych jako parametry klucza pozycji. Zwracane
              wartości są ujęte w nawiasy kwadratowe i poprzedzone typem
              zwracanej wartości, oddzielonym znakiem pionowej kreski. Dla
              parametrów użytkownika typem jest zawsze t, ponieważ agent nie
              może określić wszystkich możliwych zwracanych wartości. Pozycje
              wyświetlane jako działające nie mają gwarancji działania z serwera
              Zabbix lub zabbix_get podczas odpytywania uruchomionego demona
              agenta, ponieważ uprawnienia lub środowisko mogą być inne.
              Typy zwracanych wartości to:

           d   Liczba z częścią dziesiętną.

           m   Nieobsługiwane. Może to być spowodowane odpytywaniem pozycji,
               która działa tylko w trybie aktywnym, jak pozycja monitorowania
               logu, albo pozycji wymagającej wielu zebranych wartości. Problemy
               z uprawnieniami lub nieprawidłowe parametry użytkownika również
               mogą skutkować stanem nieobsługiwanym.

           s   Tekst. Maksymalna długość nie jest ograniczona.

           t   Tekst. To samo co s.
```

```
           u   Nieujemna liczba całkowita.

       -t, --test item-key
              Przetestuj pojedynczą pozycję i zakończ. Opis danych wyjściowych
              znajduje się przy opcji --print.

       -T, --test-config
              Zweryfikuj plik konfiguracyjny i zakończ.

       -h, --help
              Wyświetl tę pomoc i zakończ.

       -V, --version
              Wyświetl informacje o wersji i zakończ.

PLIKI
       /usr/local/etc/zabbix_agentd.conf
              Domyślna lokalizacja pliku konfiguracyjnego agenta Zabbix (jeśli
              nie została zmieniona podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       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)               Ogólny podręcznik poleceń            ZABBIX_GET(1)

NAZWA
       zabbix_get - narzędzie pobierania Zabbix

SKŁADNIA
       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

OPIS
       zabbix_get jest narzędziem wiersza poleceń do pobierania danych z
       agent Zabbix.

OPCJE
       -s, --host host-name-or-IP
              Określa nazwę hosta lub adres IP hosta.

       -p, --port port-number
              Określa numer portu agent działającego na hoście. Domyślnie jest
              to 10050.
```

```
-I, --source-address IP-address
       Określa źródłowy adres IP.

-t, --timeout seconds
       Określa limit czasu. Prawidłowy zakres: 1-600 sekund (domyślnie:
       30)

-k, --key item-key
       Określa klucz pozycji, dla której ma zostać pobrana wartość.

-P, --protocol value
       Protokół używany do komunikacji z agent. Wartości:

            auto        połącz przy użyciu protokołu JSON, w razie nie-
                        powodzenia przełącz się i ponów próbę z użyciem
                        protokołu tekstowego (domyślnie)

            json        połącz przy użyciu protokołu JSON

            plaintext   połącz przy użyciu protokołu tekstowego, w któ-
                        rym wysyłany jest tylko klucz pozycji (wersje
                        6.4.x i starsze)

--tls-connect value
       Sposób połączenia z agent. Wartości:

            unencrypted
                        połącz bez szyfrowania (domyślnie)

            psk         połącz przy użyciu TLS i klucza współdzielonego

            cert        połącz przy użyciu TLS i certyfikatu

--tls-ca-file CA-file
       Pełna ścieżka do pliku zawierającego certyfikaty nadrzędnego ur-
       zędu(ów) certyfikacji (CA) do weryfikacji certyfikatu peer.

--tls-crl-file CRL-file
       Pełna ścieżka do pliku zawierającego unieważnione certyfikaty.

--tls-agent-cert-issuer cert-issuer
       Dozwolony wystawca certyfikatu agent.

--tls-agent-cert-subject cert-subject
       Dozwolony temat certyfikatu agent.

--tls-cert-file cert-file
       Pełna  ścieżka  do  pliku zawierającego certyfikat lub łańcuch
       certyfikatów.

--tls-key-file key-file
       Pełna ścieżka do pliku zawierającego klucz prywatny.

--tls-psk-identity PSK-identity
       Ciąg tożsamości PSK.

--tls-psk-file PSK-file
       Pełna ścieżka do pliku zawierającego klucz współdzielony.

--tls-cipher13 cipher-string
       Ciąg szyfrów dla OpenSSL 1.1.1 lub nowszego dla TLS 1.3. Zastępu-
       je domyślne kryteria wyboru zestawu szyfrów. Ta opcja nie jest
```

```
              dostępna, jeśli wersja OpenSSL jest starsza niż 1.1.1.

       --tls-cipher cipher-string
              Ciąg priorytetów GnuTLS (dla TLS 1.2 i nowszych) lub ciąg szyfrów
              OpenSSL (tylko dla TLS 1.2). Zastępuje domyślne kryteria wyboru
              zestawu szyfrów.

       -h, --help
              Wyświetla tę pomoc i kończy działanie.

       -V, --version
              Wyświetla informacje o wersji i kończy działanie.

KOD ZAKOŃCZENIA
       Kod zakończenia wynosi 0, jeśli zabbix_get pomyślnie pobierze wartość z
       agent Zabbix. Jeśli pobranie wartości się nie powiedzie, kod zakończenia
       wynosi 1.

PRZYKŁADY
       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

ZOBACZ TAKŻE
       Dokumentacja

       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)               Podręcznik poleceń ogólnych               ZABBIX_JS(1)

NAZWA
       zabbix_js - narzędzie JS Zabbix

SKŁADNIA
       zabbix_js -s plik-skryptu -p parametr-wejściowy [-l poziom-logowania] [-t limit-czasu]
       zabbix_js -s plik-skryptu -i plik-wejściowy [-l poziom-logowania] [-t limit-czasu]
       zabbix_js -h
       zabbix_js -V

OPIS
       zabbix_js  jest  narzędziem  wiersza  poleceń,  którego  można  używać
       do testowania osadzonych skryptów.

OPCJE
       -s, --script plik-skryptu
              Określa nazwę pliku skryptu do wykonania. Jeśli jako nazwę pliku
              podano '-', skrypt zostanie odczytany ze stdin.

       -p, --param parametr-wejściowy
```

```
              Określa parametr wejściowy przekazywany do skryptu. Parametr jest
              przekazywany dosłownie jako ciąg znaków i jest dostępny w
              skrypcie za pośrednictwem zmiennej 'value'. Wzajemnie wykluczające
              się z -i.

       -i, --input plik-wejściowy
              Określa plik zawierający parametr wejściowy przekazywany do skry-
              ptu. Zawartość pliku jest przekazywana dosłownie jako ciąg
              znaków i jest dostępna w skrypcie za pośrednictwem zmiennej 'value'.
              Jeśli jako nazwę pliku podano '-', dane wejściowe zostaną odczytane
              ze stdin. Wzajemnie wykluczające się z -p.

       -w, --webdriver url
              Określa URL webdrivera.

       -l, --loglevel poziom-logowania
              Określa poziom logowania (zakres: 0-5).

       -t, --timeout limit-czasu
              Określa limit czasu w sekundach. Prawidłowy zakres: 1-600 sekund
              (domyślnie: 10)

       -h, --help
              Wyświetla tę pomoc i kończy działanie.

       -V, --version
              Wyświetla informacje o wersji i kończy działanie.

PRZYKŁADY
       zabbix_js -s script-file.js -p example

ZOBACZ TAKŻE
       Dokumentacja

       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)             Podręcznik administratora systemu            ZABBIX_PROXY(8)

NAZWA
       zabbix_proxy - demon proxy Zabbix

SKŁADNIA
       zabbix_proxy [-c plik-konfiguracyjny]
       zabbix_proxy [-c plik-konfiguracyjny] -R opcja-czasu-uruchomienia
       zabbix_proxy [-c plik-konfiguracyjny] -T
       zabbix_proxy -h
       zabbix_proxy -V

OPIS
       zabbix_proxy jest demonem, który zbiera dane monitorowania z urządzeń i
       wysyła je do serwera Zabbix.

OPCJE
       -c, --config plik-konfiguracyjny
              Użyj alternatywnego pliku-konfiguracyjnego zamiast domyślnego.
```

```
-f, --foreground
       Uruchom proxy Zabbix na pierwszym planie.

-R, --runtime-control opcja-czasu-uruchomienia
       Wykonaj funkcje administracyjne zgodnie z opcja-czasu-uruchomienia.

    Opcje sterowania w czasie uruchomienia
        config_cache_reload
            Przeładuj pamięć podręczną konfiguracji. Zignorowane, jeśli pamięć podręczna jest obecn

        snmp_cache_reload
            Przeładuj pamięć podręczną SNMP.

        housekeeper_execute
            Uruchom housekeeper. Zignorowane, jeśli housekeeper jest obecnie wykonywany.

        diaginfo[=sekcja]
            Zapisz wewnętrzne informacje diagnostyczne dla określonej sekcji. Sekcja może mieć wart
            Domyślnie zapisywane są informacje diagnostyczne ze wszystkich sekcji.

        prof_enable[=cel]
            Włącz profilowanie, wpływa na wszystkie procesy, jeśli cel nie jest określony

        prof_disable[=cel]
            Wyłącz profilowanie, wpływa na wszystkie procesy, jeśli cel nie jest określony

        log_level_increase[=cel]
            Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli cel nie jest określony.

        log_level_decrease[=cel]
            Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli cel nie jest określony.

    Cele sterowania poziomem logowania
        process-type
            Wszystkie procesy określonego typu (availability manager, browser poller, configuration

        process-type,N
            Typ procesu i numer (np. poller,3)

        pid Identyfikator procesu

    Cele sterowania profilowaniem
        process-type
            Wszystkie procesy określonego typu (availability manager, browser poller, configuration

        process-type,N
            Typ procesu i numer (np. history syncer,1)

        pid Identyfikator procesu

        scope
            (rwlock, mutex, processing) może być użyte z process-type
            (np. history syncer,1,processing)

-T, --test-config
       Sprawdź poprawność pliku konfiguracyjnego i zakończ.

-h, --help
       Wyświetl tę pomoc i zakończ.

-V, --version
       Wyświetl informacje o wersji i zakończ.
```

```
PLIKI
       /usr/local/etc/zabbix_proxy.conf
              Domyślna lokalizacja pliku konfiguracyjnego proxy Zabbix (jeśli nie
              została zmodyfikowana podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       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)            Podręcznik poleceń ogólnych           ZABBIX_SENDER(1)

NAZWA
       zabbix_sender - narzędzie zabbix_sender

SKŁADNIA
       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
```

OPIS

zabbix_sender to narzędzie wiersza poleceń służące do wysyłania danych monitorowania do serwera Zabbix lub proxy. Na serwerze Zabbix należy utworzyć pozycję typu Zabbix trapper z odpowiednim kluczem. Należy pamiętać, że wartości przychodzące będą akceptowane wyłącznie z hostów określonych w polu Allowed hosts tej pozycji.

OPCJE

-c, --config config-file
: Użyj config-file. zabbix_sender odczytuje szczegóły serwera z pliku konfiguracyjnego agentd. Domyślnie zabbix_sender nie odczytuje żadnego pliku konfiguracyjnego. Obsługiwane są tylko parametry Hostname, ServerActive, SourceIP, TLSConnect, TLSCAFile, TLSCRLFile, TLSServerCertIssuer, TLSServerCertSubject, TLSCertFile, TLSKeyFile, TLSPSKIdentity i TLSPSKFile. Hostname zdefiniowany za pomocą parametru HostnameItem nie zostanie pobrany; w takim przypadku nazwę hosta należy określić w wierszu poleceń (zobacz opcję -s). Wszystkie adresy zdefiniowane w parametrze konfiguracyjnym ServerActive agenta są używane do wysyłania danych. Jeśli wysyłanie danych wsadowych do jednego adresu nie powiedzie się, kolejne wsady nie są wysyłane na ten adres.

-z, --zabbix-server server
: Nazwa hosta lub adres IP serwera Zabbix. Jeśli host jest monitorowany przez proxy, należy użyć nazwy hosta lub adresu IP proxy. Użycie razem z opcją --config powoduje zastąpienie wpisów parametru ServerActive określonych w pliku konfiguracyjnym agentd.

-p, --port port
: Określ numer portu, na którym działa trapper serwera Zabbix. Wartość domyślna to 10051. Użycie razem z opcją --config powoduje zastąpienie wpisów portów parametru ServerActive określonych w pliku konfiguracyjnym agentd.

-I, --source-address IP-address
: Określ źródłowy adres IP. Użycie razem z opcją --config powoduje zastąpienie parametru SourceIP określonego w pliku konfiguracyjnym agentd.

-t, --timeout seconds
: Określ limit czasu. Dozwolony zakres: 1-300 sekund (domyślnie: 60).

-s, --host host
: Określ nazwę hosta, do którego należy pozycja (zarejestrowaną w frontend Zabbix). Adres IP hosta ani nazwa DNS nie będą działać. Użycie razem z opcją --config powoduje zastąpienie parametru Hostname określonego w pliku konfiguracyjnym agentd.

-k, --key key
: Określ klucz pozycji, do której ma zostać wysłana wartość.

-o, --value value
       Określ wartość pozycji.

-i, --input-file input-file
       Wczytaj wartości z pliku wejściowego. Określ - jako <input-file>,
       aby odczytywać wartości ze standardowego wejścia. Każdy wiersz
       pliku zawiera elementy rozdzielone białymi znakami:
       <hostname> <key> <value>. Każda wartość musi być określona
       w osobnym wierszu. Każdy wiersz musi zawierać 3 elementy
       rozdzielone białymi znakami: <hostname> <key> <value>, gdzie
       „hostname" to nazwa monitorowanego hosta zarejestrowana
       w frontend Zabbix, „key" to docelowy klucz pozycji, a „value"
       to wartość do wysłania. Określ - jako <hostname>, aby użyć nazwy
       hosta z pliku konfiguracyjnego agenta lub z argumentu --host.

       Przykład wiersza pliku wejściowego:

        "Linux DB3" db.connections 43

       Typ wartości musi być prawidłowo ustawiony w konfiguracji pozycji
       w frontend Zabbix. zabbix_sender wyśle do 250 wartości w ramach
       jednego połączenia. Zawartość pliku wejściowego musi być zakodowana
       w UTF-8. Wszystkie wartości z pliku wejściowego są wysyłane
       sekwencyjnie, od góry do dołu. Wpisy muszą być sformatowane zgodnie
       z następującymi zasadami:

              • Obsługiwane są wpisy ujęte w cudzysłowy i nieujęte
                w cudzysłowy.

              • Znakiem cytowania jest cudzysłów.

              • Wpisy zawierające białe znaki muszą być ujęte
                w cudzysłowy.

              • Znaki cudzysłowu i ukośnika odwrotnego wewnątrz wpisu
                ujętego w cudzysłowy muszą być poprzedzone ukośnikiem
                odwrotnym.

              • Wpisy nieujęte w cudzysłowy nie obsługują znaków
                ucieczki.

              • Sekwencje ucieczki znaku nowego wiersza (\n) są
                obsługiwane w ciągach ujętych w cudzysłowy.

              • Sekwencje ucieczki znaku nowego wiersza są usuwane
                z końca wpisu.

-T, --with-timestamps
       Tej opcji można używać wyłącznie z opcją --input-file.

       Każdy wiersz pliku wejściowego musi zawierać 4 elementy rozdzielone
       białymi znakami: <hostname> <key> <timestamp> <value>. Znacznik
       czasu należy określić w formacie znacznika czasu Unix. Jeśli
       docelowa pozycja ma odwołujące się do niej wyzwalacze, wszystkie
       znaczniki czasu muszą być rosnące, w przeciwnym razie obliczanie
       zdarzeń nie będzie prawidłowe.

       Przykład wiersza pliku wejściowego:

        "Linux DB3" db.connections 1429533600 43

       Więcej informacji zawiera opis opcji --input-file.

Jeśli wartość ze znacznikiem czasu zostanie wysłana dla hosta, który znajduje się w okresie utrzymania typu „no data", wartość ta zostanie odrzucona; można jednak wysłać wartość ze znacznikiem czasu przypadającym na zakończony okres utrzymania i zostanie ona zaakceptowana.

-N, --with-ns
Tej opcji można używać wyłącznie z opcją --with-timestamps.

Każdy wiersz pliku wejściowego musi zawierać 5 elementów rozdzielonych białymi znakami: <hostname> <key> <timestamp> <ns> <value>.

Przykład wiersza pliku wejściowego:

"Linux DB3" db.connections 1429533600 7402561 43

Więcej informacji zawiera opis opcji --input-file.

-r, --real-time
Wysyłaj wartości pojedynczo, natychmiast po ich otrzymaniu. Można tego używać podczas odczytu ze standardowego wejścia.

--tls-connect value
Sposób łączenia z serwerem lub proxy. Wartości:

unencrypted
połącz bez szyfrowania (domyślnie)

psk
połącz za pomocą TLS i klucza wstępnie współdzielonego

cert
połącz za pomocą TLS i certyfikatu

--tls-ca-file CA-file
Pełna ścieżka do pliku zawierającego certyfikaty najwyższego poziomu CA do weryfikacji certyfikatu peera.

--tls-crl-file CRL-file
Pełna ścieżka do pliku zawierającego unieważnione certyfikaty.

--tls-server-cert-issuer cert-issuer
Dozwolony wystawca certyfikatu serwera.

--tls-server-cert-subject cert-subject
Dozwolony podmiot certyfikatu serwera.

--tls-cert-file cert-file
Pełna ścieżka do pliku zawierającego certyfikat lub łańcuch certyfikatów.

--tls-key-file key-file
Pełna ścieżka do pliku zawierającego klucz prywatny.

--tls-psk-identity PSK-identity
Ciąg identyfikatora PSK.

--tls-psk-file PSK-file
Pełna ścieżka do pliku zawierającego klucz wstępnie współdzielony.

--tls-cipher13 cipher-string
Ciąg szyfru dla OpenSSL 1.1.1 lub nowszego dla TLS 1.3.

Zastępuje domyślne kryteria wyboru zestawu szyfrów. Ta opcja nie jest dostępna, jeśli wersja OpenSSL jest starsza niż 1.1.1.

--tls-cipher cipher-string
Ciąg priorytetów GnuTLS (dla TLS 1.2 i nowszych) lub ciąg szyfru OpenSSL (tylko dla TLS 1.2). Zastępuje domyślne kryteria wyboru zestawu szyfrów.

-v, --verbose
Tryb szczegółowy; -vv wyświetla więcej informacji.

-h, --help
Wyświetl tę pomoc i zakończ działanie.

-V, --version
Wyświetl informacje o wersji i zakończ działanie.

KOD ZAKOŃCZENIA

Kod zakończenia wynosi 0, jeśli wartości zostały wysłane i wszystkie zostały pomyślnie przetworzone przez serwer. Jeśli dane zostały wysłane, ale przetwarzanie co najmniej jednej wartości nie powiodło się, kod zakończenia wynosi 2. Jeśli wysyłanie danych nie powiodło się, kod zakończenia wynosi 1.

PRZYKŁADY

zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -k mysql.queries -o 342.45
Wyślij 342.45 jako wartość pozycji mysql.queries monitorowanego hosta. Użyj monitorowanego hosta i serwera Zabbix zdefiniowanych w pliku konfiguracyjnym agenta.

zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -s "Monitored Host" -k mysql.queries -o 342.45
Wyślij 342.45 jako wartość pozycji mysql.queries hosta Monitored Host, używając serwera Zabbix zdefiniowanego w pliku konfiguracyjnym agenta.

zabbix_sender -z 192.168.1.113 -i data_values.txt
Wyślij wartości z pliku data_values.txt do serwera Zabbix o adresie IP 192.168.1.113. Nazwy hostów i klucze są zdefiniowane w pliku.

echo "- hw.serial.number 1287872261 SQ4321ASDF" | zabbix_sender -c /usr/local/etc/zabbix_agentd.conf -T -i -
Wyślij wartość ze znacznikiem czasu z wiersza poleceń do serwera Zabbix określonego w pliku konfiguracyjnym agenta. Myślnik w danych wejściowych oznacza, że nazwa hosta również powinna zostać użyta z tego samego pliku konfiguracyjnego.

echo '"Zabbix server" trapper.item ""' | zabbix_sender -z 192.168.1.113 -p 10000 -i -
Wyślij pustą wartość pozycji do serwera Zabbix o adresie IP 192.168.1.113 na porcie 10000 z wiersza poleceń. Puste wartości należy oznaczać pustymi cudzysłowami.

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
Wyślij 342.45 jako wartość pozycji mysql.queries hosta Monitored Host do serwera o adresie IP 192.168.1.113, używając TLS z certyfikatem.

```
       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
              Wyślij 342.45 jako wartość pozycji mysql.queries hosta Monitored
              Host do serwera o adresie IP 192.168.1.113, używając TLS
              z kluczem wstępnie współdzielonym (PSK).

ZOBACZ TAKŻE
       Dokumentacja

       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)            Podręcznik administratora systemu           ZABBIX_SERVER(8)

NAZWA
       zabbix_server - demon serwera Zabbix

SKŁADNIA
       zabbix_server [-c plik-konfiguracyjny]
       zabbix_server [-c plik-konfiguracyjny] -R opcja-czasu-rzeczywistego
       zabbix_server [-c plik-konfiguracyjny] -T
       zabbix_server -h
       zabbix_server -V

OPIS
       zabbix_server jest głównym demonem oprogramowania Zabbix.

OPCJE
       -c, --config plik-konfiguracyjny
              Użyj alternatywnego pliku konfiguracyjnego zamiast domyślnego.

       -f, --foreground
              Uruchom serwer Zabbix na pierwszym planie.

       -R, --runtime-control opcja-czasu-rzeczywistego
              Wykonaj funkcje administracyjne zgodnie z opcją-czasu-rzeczywistego.

           Opcje sterowania w czasie rzeczywistym
               config_cache_reload
                   Przeładuj pamięć podręczną konfiguracji. Ignorowane, jeśli pamięć podręczna jest obecni

               snmp_cache_reload
                   Przeładuj pamięć podręczną SNMP.

               housekeeper_execute
                   Uruchom housekeeper. Ignorowane, jeśli housekeeper jest obecnie wykonywany.

               trigger_housekeeper_execute
                   Uruchom trigger housekeeper (usuń problemy dla usuniętych wyzwalaczy). Ignorowane, jeśl

               secrets_reload
                   Przeładuj sekrety z Vault. Ignorowane, jeśli pamięć podręczna konfiguracji lub sekrety

               diaginfo[=section]
                   Zapisz wewnętrzne informacje diagnostyczne dla określonej sekcji. Sekcja może być histo
```

```
    prof_enable[=target]
        Włącz profilowanie, wpływa na wszystkie procesy, jeśli target nie jest określony

    prof_disable[=target]
        Wyłącz profilowanie, wpływa na wszystkie procesy, jeśli target nie jest określony

    ha_status
        Wyświetl status klastra wysokiej dostępności. Można wykonać tylko na aktywnym węźle.

    ha_remove_node
        ha_remove_node[=name|ID] Usuń węzeł wysokiej dostępności według jego nazwy lub ID podan

    ha_set_failover_delay[=delay]
        Ustaw opóźnienie przełączenia awaryjnego wysokiej dostępności. Opóźnienie można określi

    log_level_increase[=target]
        Zwiększ poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony

    log_level_decrease[=target]
        Zmniejsz poziom logowania, wpływa na wszystkie procesy, jeśli target nie jest określony

Cele sterowania poziomem logowania
    process-type
        Wszystkie procesy określonego typu (alerter, alert manager,
        alert syncer, availability manager, browser poller, config-
        uration syncer, configuration syncer worker, connector man-
        ager, connector worker, discovery manager, escalator, ha m-
        anager, history poller, history syncer, housekeeper, h-
        ttp poller, http agent poller, icmp pinger, internal poll-
        er, ipmi manager, ipmi poller, java poller, lld manager,
        lld worker, odbc poller, poller, agent poller, preprocessi-
        ng 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, unreac-
        hable poller, vmware collector)

    process-type,N
        Typ procesu i numer (np. poller,3)

    pid Identyfikator procesu

Cele sterowania profilowaniem
    process-type
        Wszystkie procesy określonego typu (alerter, alert manager,
        alert syncer, availability manager, browser poller, config-
        uration syncer, configuration syncer worker, connector man-
        ager, connector worker, discovery manager, escalator, ha
        manager, history poller, history syncer, housekeeper,
        http poller, http agent poller, icmp pinger, internal po-
        ller, ipmi manager, ipmi poller, java poller, lld manager,
        lld worker, odbc poller, poller, agent poller, preproc-
        essing manager, proxy poller, proxy group manager, repo-
        rt manager, report writer, self-monitoring, service mana-
        ger, snmp poller, snmp trapper, task manager, timer, trap-
        per, trigger housekeeper, unreachable poller, vmware colle-
        ctor)

    process-type,N
        Typ procesu i numer (np. history syncer,1)

    pid Identyfikator procesu
```

```
               scope
                   (rwlock, mutex, processing) może być użyte z process-type
                   (np. history syncer,1,processing)

       -T, --test-config
              Sprawdź poprawność pliku konfiguracyjnego i zakończ.

       -h, --help
              Wyświetl tę pomoc i zakończ.

       -V, --version
              Wyświetl informacje o wersji i zakończ.

PLIKI
       /usr/local/etc/zabbix_server.conf
              Domyślna lokalizacja pliku konfiguracyjnego serwera Zabbix (jeśli
              nie została zmodyfikowana podczas kompilacji).

ZOBACZ TAKŻE
       Dokumentacja

       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)       Podręcznik administratora systemu      ZABBIX_WEB_SERVICE(8)

NAZWA
       zabbix_web_service - usługa web Zabbix

SKŁADNIA
       zabbix_web_service [-c config-file]
       zabbix_web_service [-c config-file] -T
       zabbix_web_service -h
       zabbix_web_service -V

OPIS
       zabbix_web_service jest aplikacją służącą do udostępniania usług web
       komponentom Zabbix.

OPCJE
       -c, --config config-file
              Używa alternatywnego pliku config-file zamiast domyślnego.

       -T, --test-config
              Weryfikuje plik konfiguracyjny i kończy działanie.

       -h, --help
              Wyświetla tę pomoc i kończy działanie.

       -V, --version
              Wyświetla informacje o wersji i kończy działanie.

PLIKI
       /usr/local/etc/zabbix_web_service.conf
              Domyślna lokalizacja pliku konfiguracyjnego usługi web Zabbix
              (jeśli nie została zmodyfikowana podczas kompilacji).
```

```
ZOBACZ TAKŻE
       Dokumentacja

       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)
```

Informacja o prawach autorskich

Dokumentacja Zabbix NIE jest rozpowszechniana na licencji AGPL-3.0. Korzystanie z dokumentacji Zabbix podlega następującym warunkom:

Możesz utworzyć drukowaną kopię tej dokumentacji wyłącznie do własnego użytku osobistego. Konwersja do innych formatów jest dozwolona, o ile rzeczywista treść nie zostanie w żaden sposób zmieniona ani zredagowana. Nie wolno publikować ani rozpowszechniać tej dokumentacji w żadnej formie ani na żadnym nośniku, chyba że rozpowszechniasz dokumentację w sposób podobny do tego, w jaki rozpowszechnia ją Zabbix (to znaczy elektronicznie do pobrania na stronie internetowej Zabbix) albo na nośniku USB lub podobnym, pod warunkiem jednak, że dokumentacja jest rozpowszechniana razem z oprogramowaniem na tym samym nośniku. Jakiekolwiek inne wykorzystanie, takie jak jakiekolwiek rozpowszechnianie drukowanych kopii lub wykorzystanie tej dokumentacji, w całości lub w części, w innej publikacji, wymaga uprzedniej pisemnej zgody upoważnionego przedstawiciela Zabbix. Zabbix zastrzega sobie wszelkie prawa do tej dokumentacji, które nie zostały wyraźnie przyznane powyżej.